LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-663834

Final Report: 11-SI-006 Creating Optimal Fracture Networks

F. J. Ryerson

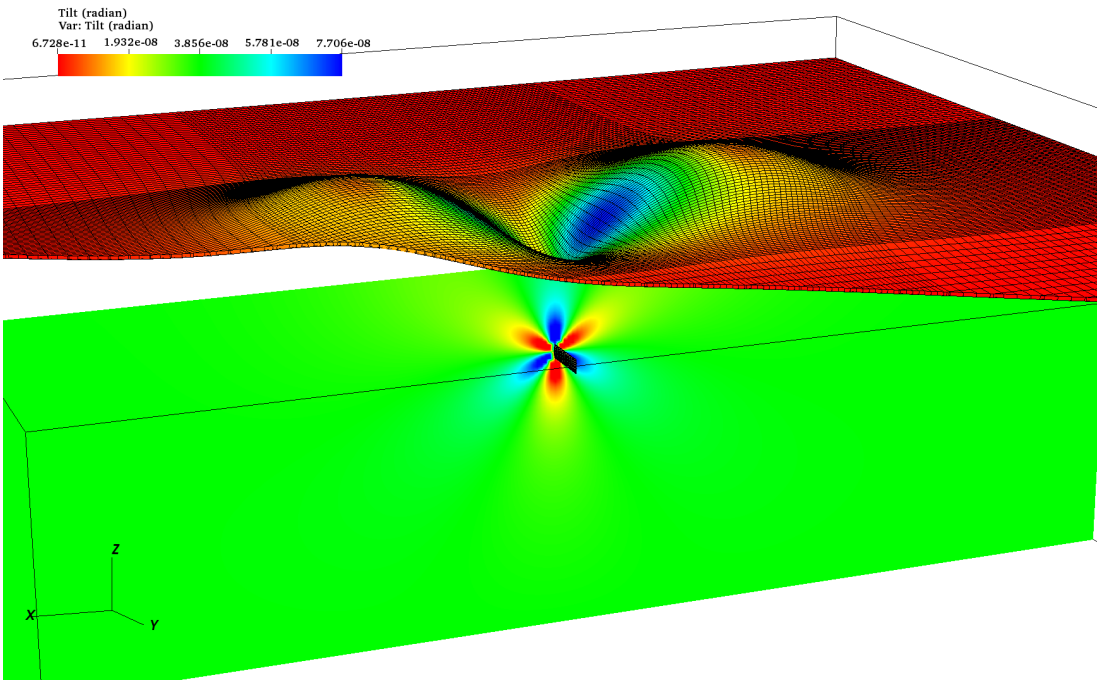
November 5, 2014

Disclaimer

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Final Report: 11-SI-006
Creating Optimal Fracture Networks

Frederick J. Ryerson
AEED/PLS
LLNL



GEOS 3D simulation of surface deformation for synthetic tiltmeter (shown in the color bar in units of radians) and INSAR measurements of subsidence and uplift due to a pressurized fracture in the subsurface.

*This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Table of Contents

Table of Contents **2**

1. Introduction: **3**

2. GEOS Description: **4**

3. Mission and Strategy: **7**

4. GEOS Development History **9**

3.1 The GEOS framework **11**

3.2 GEOS Model Enhancements **12**

3.2.1 Fully-Coupled Numerical Methods 13

3.2.2 Proppant Modeling 16

3.2.3 Flow and Transport Solver 17

3.2.4 Wave Propagation Solver 18

3.2.5 Material Modeling 18

3.2.6 Other Developments 19

4. Future Work **19**

4.1 Multi-scale Damage Modeling **20**

4.2 Uncertainty Quantification **21**

4.3 High Strain Rate Fracturing **21**

4.3.1 Deflagration-Induced Pressure Elevation in Fracture Network 21

4.3.2 Detonation-Induced Pressure Pulse in Fracture Network 22

4.3.3 Shock Wave Interaction Induced Tensile Damage Zones 22

Appendix A: Work-flow **23**

Appendix B: Quality Assurance **24**

Appendix C: Publications **25**

1. Introduction:

The nation's energy security is largely dependent on the acquisition of baseload power, with minimal environmental impact and cost. This issue constitutes one of the outstanding problems of the millennium. The solid Earth provides two primary energy sources – fossil fuels and geothermal energy. In both cases, "conventional" resources are those that can be economically harvested with existing technology. Relatively rapid fluid flow through porous, high permeability rocks makes this possible. Conventional oil recovery and hydrothermal energy are prime examples. Unconventional resources are those too costly to extract with current methods and include natural gas and oil contained in shales (shale-gas and shale-oil) and thermal energy contained deeper in the earth's crust (>3 km, these are referred to as Enhanced Geothermal Systems (EGS)). Both are rendered unconventional due to low reservoir permeability that prevents gases from flowing and geothermal "working fluids" from circulating and extracting heat. The key to efficient extraction of energy from these resources is the engineered enhancement of fracture permeability, *the creation of optimal fracture networks*.

The process of enhancing the fracture permeability of reservoir, known as "fracking", involves hydraulic "stimulation" to produce new fractures or reactivate existing fracture systems. The behavior of fracture networks is complex, exhibiting dependence on *in situ* heterogeneities, mineralogy and constitutive properties of the native rock, and on loading conditions, and influenced by the *in situ* stress field and the dynamically varying local stresses near both created and *in situ* heterogeneities. Further complications arise due to uncertainties—aleatoric and epistemic—resulting from not only inadequate characterization of the geologic environment but also from a lack of understanding of important physical processes that influence fracture initiation and propagation. Empirical engineering of fracture networks is expensive and in the case of EGS, has not resulted in a useable domestic energy supply in spite of its transformative volume and broad geographic distribution.

The primary goal of this project is the development of a multi-physics, multi-scale simulation capability to predict the initiation, propagation and maintenance of hydraulically driven fracture networks in heterogeneous geologic materials. The development and validation of integrated code capability, GEOS, that accurately captures the phenomenology of these complex physical processes across a wide range of time- (~10^{-6} to ~10^{8} s) and length- (~10^{-3} to ~10^{4} m) scales is the main scientific objective of the proposed initiative. GEOS constitutes a new analytical framework incorporating the underlying physics that drives fracture development

– *e.g*. dynamically changing fracture network topologies, fluid/solid interactions taking place along discrete interfaces, and complex matrix/fracture transport processes. Enhanced seismic observational tools and inversion methods will also be developed.

This initiative has positioned LLNL to play a leading role in the development of Enhanced Geothermal Systems, unconventional gas energy resources, risk assessment associated with geologic sequestration of CO_2 and the detection of radioactive gases from clandestine nuclear tests, impacting both energy and national security.

2. GEOS Description:

GEOS is a massively parallel, multi-physics simulation application utilizing high performance computing (HPC) and designed to address subsurface reservoir stimulation activities for the purpose of optimizing current operations and evaluating innovative stimulation methods.. GEOS enables coupling of different solvers associated with the various physical processes occurring during reservoir stimulation in unique and sophisticated ways, adapted to various geologic settings, materials and stimulation methods, and provides a platform for integrating various geophysical and operational observations during reservoir stimulation. Developed under the auspices of Lawrence Livermore National Laboratory's (LLNL) Laboratory-Directed Research and Development (LDRD) program as a Strategic Initiative (SI), GEOS represents the culmination of a three-year code development and improvement plan that has leveraged existing code capabilities and staff expertise to design new computational geosciences software.

The overall architecture of the GEOS framework includes consistent data structures, generalized parallel communication and input/output functions, and interfaces for incorporating additional physics solvers and materials models as demanded by future applications. Along with predicting the initiation, propagation and reactivation of fractures (Figure 1), GEOS also generates synthetic microseismic source terms that can be used to generate motions at surface and downhole array positions (Figure 2). GEOS can also be linked with existing, non-intrusive uncertainty quantification schemes, such as PSUADE, to constrain uncertainty in its predictions and sensitivity to the various parameters describing the reservoir and stimulation operations.

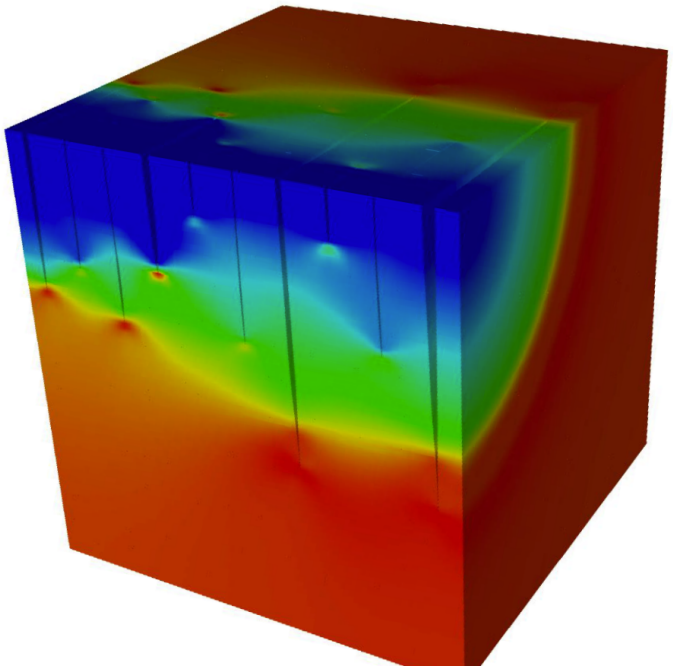


Figure 1: Illustration of nine simultaneously pumped hydraulically driven fractures, showing how small change in the fluid boundary conditions can lead to race conditions. The color bar indicates stress (blue is compressive) perpendicular to the fractures.

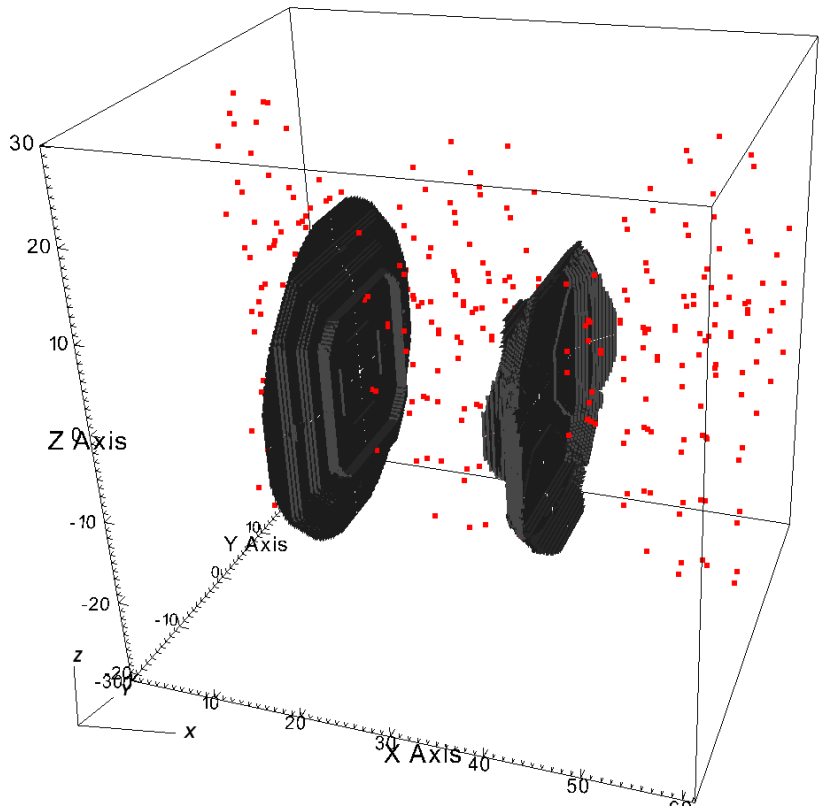


Figure 2: Illustration of two simulataneously pumped fractures under a slight pressure gradient. The fractures curve away from each other in three dimensions with the higher pressure fracture deflecting less than the other. The red dots indicate pre-existing joints in the simulation where microseismicity is predicted.

GEOS development was originally motivated by the need to simulate hydraulic fracture stimulation; however, the capabilities are being expanded beyond this to include simulation of long-term fault behavior associated with injection-induced/triggered seismicity (Figure 3), modeling the behavior of discontinuous rock masses under load, particle method simulation of granular mechanics, flow and transport of heat and fluid in dual permeability (discontinuous/continuous) geologies, and numerical (finite difference based) propagation of seismic waves. GEOS can additionally call LLNL-proprietary material models and equations of state.

Despite the expansion of its application space, the GEOS framework development is

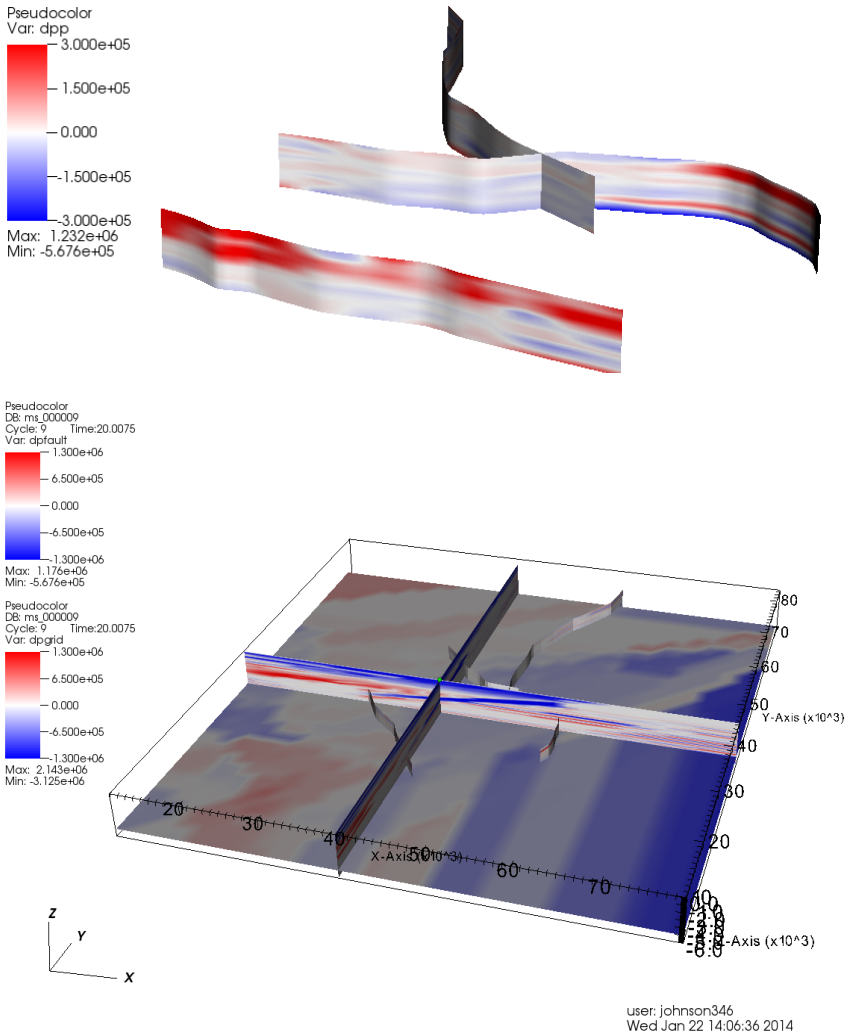


Figure 3: Pore pressure perturbation along pre-existing faults (upper) and the within slices through the reservoir (lower) for a simulation (GEOS coupled with PNNL's STOMP reservoir simulator) of seismicity induced by CO2 injection in a seismically active region. Fault slip is explicitly tracked using a Displacement Discontinuity Method (DDM) to resolve the mechanics and a rate-and-state based formulation to inform the failure criterion in the same way as RSQSim (Richards-Dinger and Dieterich, UC-Riverside)

primarily focused on the solution of low-rate loading of coupled hydro-mechanical systems with the target application of better characterizing reservoir response to different stimulation, fracture control, and injection/disposal techniques over both stimulation and production timescales.

3. Mission and Strategy:

Due to the vast range of relevant time and spatial domains, coupled with uncertain boundary conditions, the development of numerical simulation tools to address problems in the geological sciences and geologic engineering applications represents a continuing challenge. It has been difficult to develop effective numerical tools that incorporate accurate models of the relevant physical mechanisms and the requisite spatial and temporal scaling, while maintaining computational economy. Instead, many tools have been developed utilizing field-based experience and/or empirical relationships, and can work extraordinarily well when the operation is mature and a significant operational database exists. These methods, however, work less well when one needs to predict the outcome of a new operational method or work in a reservoir that has geologic characteristics different from those that are familiar. Such complexities are encountered in unconventional gas and oil production, development of enhanced geothermal systems and in geologic carbon sequestration – each an important energy security and environmental concern. However, it is these new operational methods and unfamiliar geologies that hold the greatest prospect for unlocking new energy resources and increasing the efficiency of resources extraction.

Currently, the oil and gas industry has been able to address hydraulic stimulation operations from a largely empirical standpoint, without resorting to sophisticated computational tools. The advent of slick-water hydraulic fracture stimulation coupled with horizontal drilling and staged fracturing, the technologies that have spurred the current boom in tight gas drilling, is an example of such an advance. This is characteristic of many advances, where trial-and-error over many projects by many operators eventually results in a technique that works. However, over 20 years elapsed between the initial federal R&D investment in shale-gas extraction and the first economical shale-gas fracture. Similarly, this empirical approach can be inefficient in terms of resource allocation (the cost of drilling a single well can be ~$1M) and may also have significant environmental risks. As the implementation of hydraulic stimulation expands to encompass different lithologies (e.g., the Monterey Formation) and stress regimes, this empirical approach may no longer be relevant. The primary issue addressed by the GEOS Initiative is whether or not HPC-based simulation of extraction methods can be applied to a variety of geologic models and be used to optimize existing extraction operations and to evaluate the efficacy of innovative methodologies.

Given the ever-increasing need for energy with minimal environmental impact, the prospects of HPC-based simulation tools in optimizing operations during both planning and execution is tantalizing, with potentially drastic decreases in cost, faster

turnaround, greater control of the analysis, and more complete utilization of operational data. However, this potential is only realized if the simulations can demonstrably accelerate the deployment of new methodologies and predict system responses within reliable error bounds. This would not only improve operations and efficiency but also decrease project failure risk, which has far-reaching implications from insurance valuation to government regulation. Everyone benefits when risk can be reliably assessed and reduced.

The major hurdle in developing such numerical simulation tools lies in the complex physical processes at work in the reservoir, along with the uncertain nature of reservoir characteristics. Processes associated with stimulation vary drastically in time and spatial extent, and the dominant physical processes at any location within a reservoir may also vary as stimulation proceeds. For instance, while the timescale for long-term production is on the order of decades, a stimulation operation may take less than a day, and individual fractures may propagate via processes at the microsecond timescale if viewed at the smallest spatial scales. The spatial domain also varies drastically: Reservoirs may be on the order of less than one to several miles, while a particular stimulation may only affect an area of several tens of feet from the action point. The choice of stimulation method may additionally influence the time and spatial domains of resulting physical phenomena that must be considered.

GEOS is a framework that allows an analyst to couple different solvers, adapting to various geologic settings, materials and stimulation methods. Currently, GEOS development is focused on capturing the dominant processes during stimulation of fracture networks in the subsurface. This problem is uniquely addressable with GEOS, which can run massively parallelized solvers (sequentially/concurrently) with different coupling interfaces. The envisioned end use case would involve using an implicit solver for the majority of the simulation at a coarse scale, allowing for fast computation of the evolution of the reservoir. When a fracturing event is imminent, the solver is transitioned to a more appropriate explicit solver that captures the time evolution of the event then synchronizes with the implicit solver to continue its simulation.

Coupling between physics models occurs in a similar manner. For most of the time the hydraulic behavior of rock with pre-existing joints and fractures is governed by slow, convective, laminar flow through narrow through-going channels and by even slower diffusive and/or Darcy flow in the porous rock. However, during stimulation these time constants can be drastically reduced and non-laminar effects can dominate in the fractures where fracture propagation is occurring. GEOS allows coupling of fracture evolution with a seismic response model to determine the source mechanism associated with each event. This, in turn, can be coupled with other wave propagation codes (and will be addressed by a finite difference code currently under development directly in GEOS) to predict the synthetic micro-seismicity at a particular site. Similarly, a simulated anisotropic permeability tensor can be handed off to fluid and thermal transport codes (and will be addressed by a dual-permeability finite element code

currently under development directly in GEOS). In the end, this scheme will provide a clear and consistent upscaling method to take the mechanisms that are observed at finer scales and homogenize them to larger scales for specific sites, providing the additional ability to capture uncertainty in the calculations as well as providing the basis for comprehensive risk analysis.

4. GEOS Development History

The GEOS Initiative is a 3-year project funded by LLNL's Laboratory Directed Research and Development program (LDRD) beginning in July 2011. Here we describe the essential elements of the GEOS code and their implementation.

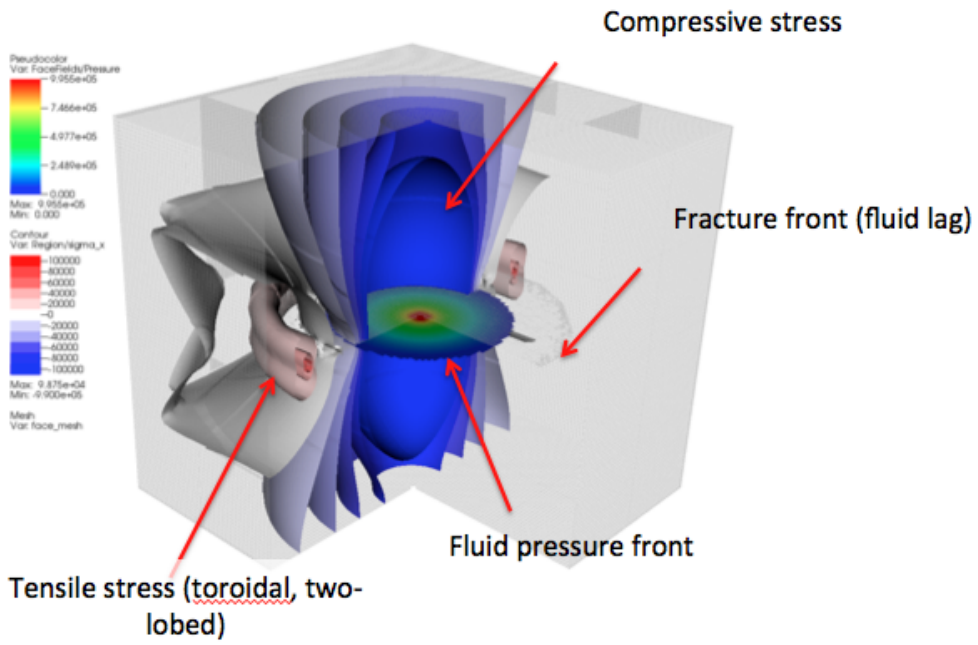


Figure 4: Illustration of a single, penny-shaped (radial) fracture propagating. The tip stress at the leading edge of the fracture is indicated as a darker gray ring with the (light red) tensile stress while the fluid lags the fracture front.

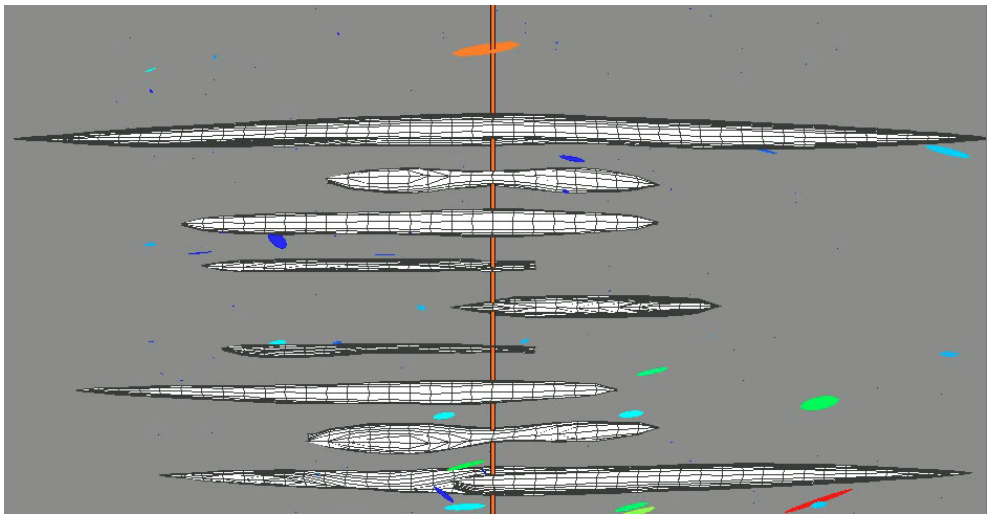


Figure 5: Illustration of simultaneously pumped hydraulic fractures in a randomly perturbed matrix and under differential stress. Microseismic source tensors induced by the changing stress field near the hydraulic fractures are rendered as oriented, scaled ellipsoids and colored by event magnitude. Note that the seismicity associated with the primary hydraulic fracture has been suppressed to highlight events in the matrix related to perturbations in local stresses that are unrelated to the fluid front.

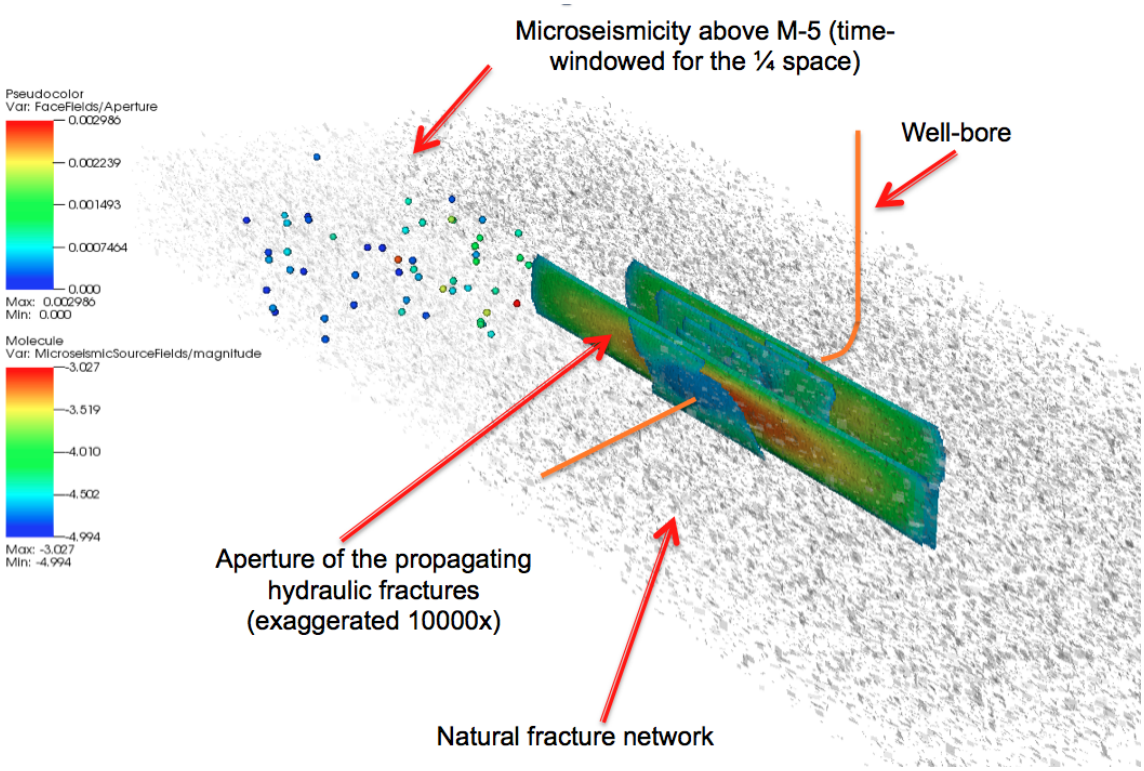


Figure 6: Illustration of a simulaneously fractured stage in the presence of inhomogeneities. The pre-existing fracture network for calculating the failure state is shown as gray rectangles, and calculated microseisms are shown in the left, upper quadrant.

3.1 The GEOS framework

GEOS currently supports two- and three-dimensional coupled finite difference (FD) and finite element (FE) based simulation:

- First-order accurate elements for FE solid mechanics
    - 2D: triangular and quadrilateral elements
    - 3D: tetrahedral and hexahedral elements
- Dimensionally reduced, first-order accurate elements for FD parallel plate idealized fluid dynamics
    - 1D: line elements
    - 2D: triangular and quadrilateral elements
- Proppant models include several different fluid rheologies:
    - Non-Newtonian
    - Proppant entrained
    - Multi-component settling and transport
- Fracture mechanics is addressed at multiple scales:
    - Fine scale: typical cohesive element approach
    - Coarse-scale: displacement-based approach to calculated the stress intensity factor at the fracture tip
- When the fracture criterion has been met, new surface area is created:
    - Solid mesh topology is changed through the separation of faces
    - Fluid mesh topology is changed through the spawning of new, fluid mesh elements that conform to the faces of the newly created surface area
    - Connectivity between the solid and fluid elements is established and coupling achieved through the pressure term in the fluid and the displacement term in the solid.
- If fracture surfaces come back into contact, this condition is automatically detected:
    - Several contact detection algorithms implemented, including algorithms that scale linearly in computational time with the number of interfaces (i.e., O(N) complexity algorithms)
    - Large-displacement of surfaces is handled through an advection-free algorithm
    - Arbitrary interfacial constitutive models can inform the permeability-displacement and traction-displacement relations
        - For example: Barton-Bandis-Bakhtar type models are implemented
- In-situ fractures and non-linear, anisotropic behavior can be included in the mechanical and seismic response:
    - Anisotropic damage models are currently available with LLNL proprietary material libraries
    - A new anisotropic elasticity and anisotropic plasticity model aimed at shale behavior is currently being implemented

- Micro-seismic failure models, which provide full tensor source terms, are supported in the code
  - Support user input or geostatistically populated fractures and fracture properties (e.g., cohesion, rate-and-state friction parameters, etc.)
  - In-situ fracture state is uni-directionally coupled with the in-situ stress state in the FE solid elements
  - Two-way coupling such that the fractures inform the material properties of the solid is under development concurrent with the development of the anisotropic elasto-plastic constitutive model.

3.2 GEOS Model Enhancements

While the essential elements of the GEOS are substantially complete, we continue to optimize code performance and computational efficiency, while also incorporating additional physical models. Many of these improvements are the result of interactions with external sponsors who are working with LLNL on GEOS applications to various hydrocarbon extraction issues. The following elements are currently in being implemented:

- Finite element based hydrothermal flow and transport solver development
  - Completed initial implementation
    - Scalable
    - Massively parallel
  - Discontinuous Galerkin (DG) version under development to handle strong pressure gradients near the interface for fractured, low matrix permeability shales.
- Finite difference based seismic wave propagation solver development
  - Completed initial implementation
    - Isotropic elasticity formulation
    - Cross-code validation against SW4
  - Anisotropic elasticity version under development to handle bedded geologies and transversely anisotropic material response.
- Discrete element solver development for large deformation, massively parallel applications
  - Dynamic repartitioning under development

A longer-term goal of the GEOS development effort,is the development of derivative, fast-running, reduced complexity codes that can be traced back and verified using the full GEOS code. In the interim, we maintain a 2D modeling capabilities in GEOS that can be utilized in reconnaissance evaluation, plane strain verification and validation (V&V), and in uncertainty quantification and sensitivity analysis. The 2D implementation also provides a simpler visualization of the physical processes associated with stimulation.

3.2.1 Fully-Coupled Numerical Methods

Work has been completed for the explicitly integrated case, which relies on the principles of "dynamic relaxation" to achieve a quasi-static solution of the stiff equations governing the coupled hydro-mechanical system. However, more development is currently being undertaken to improve robustness and decrease computational intensity. We have recently completed the development of the fully-coupled implicit solution as a more accurate alternative to the explicit approach.

3.2.1.1 Explicit Developments

"Dynamic relaxation" uses an explicit time step with damping and increased inertial terms (i.e., density) to converge to the equilibrium solution rather than using implicit iteration. Such an approach can be more robust for highly nonlinear systems but requires that one make a sufficiently accurate initial guess during each time step (since there is no iteration or correction). When we have no confinement, the approach lets us use trivial guesses for the zero-stress initial fluid and contact states in newly created surface area; because we have accurate initial guesses, and the further evolution of the system is insensitive to fluid and contact stiffness, we can reduce the system stiffness (and thereby use longer time steps to reduce computational effort) with only modest (and bounded) increases in error.

However, such trivial guesses, when used as the initial guess for non-zero stress state case, become increasingly inaccurate as confinement increases, requiring us currently to use the more accurate (and stiffer) fluid and contact properties to achieve the same accuracy. This has the effect of driving down the time step necessary to relax the system in a finite time period and drastically increases the computational effort.

We are developing more appropriate initialization schemes for the fluid and contact states associated with new areas. The logic for this has been implemented in a development[1] branch of the code, but it has yet to be migrated into the production branch that is currently being used to run the verification problems.

3.2.1.2 Implicit Developments

The more established method for solving general, fully-coupled systems is through iterative solution using implicit schemes. For this approach, we have developed a block-solution approach to solve the system of coupled equations. The blocks are separated into diagonal and off-diagonal blocks. The diagonal blocks are where the matrices for particular atomic solvers (i.e., a solver associated with a particular subset of the degrees-of-freedom) are specified; for instance, the stiffness matrix associated with the solid mechanics displacements may comprise a block while the permeability matrix associated with the fluxes may comprise another. The off-diagonal blocks are where the matrices associated with the coupling between atomic solvers are specified; for

[1] For descriptions of the "production" and "development" branch distinctions, please see Appendix B

instance, the off-diagonal blocks may describe how the permeability depends on displacements or the stiffness depends on fluxes.

The resultant system of equations is quite stiff due to the relationship between interfacial displacements and (1) the permeability on the of the solid along fracture boundaries, which exhibits cubic dependence and (2) the solid stiffness of the fluid within the region between two fracture faces, which is characterized by small dimensions and a non-linear, strain-stiffening modulus. Progress this quarter, however, has included the implementation of block pre-conditioners to reduce the computational cost of evaluating the system of equations and increased the robustness of the solution..

3.2.1.3 Contact

We have previously implemented models for a sub-fracture resolution of permeability (Barton-Bandis-Bakhter) alongside the LDEC model (Livermore Discrete Element Code) for joint dilation and normal response in the GEOS framework, enabling simulation of dilating, rough fractures. We have also implemented a more advanced friction model (i.e., Dieterich-Ruina rate-and-state friction) along with empirical relationships between the parameters of these models and the total organic content and clay weight (TOCC) of source rocks as developed from data by Kohli and Zoback.

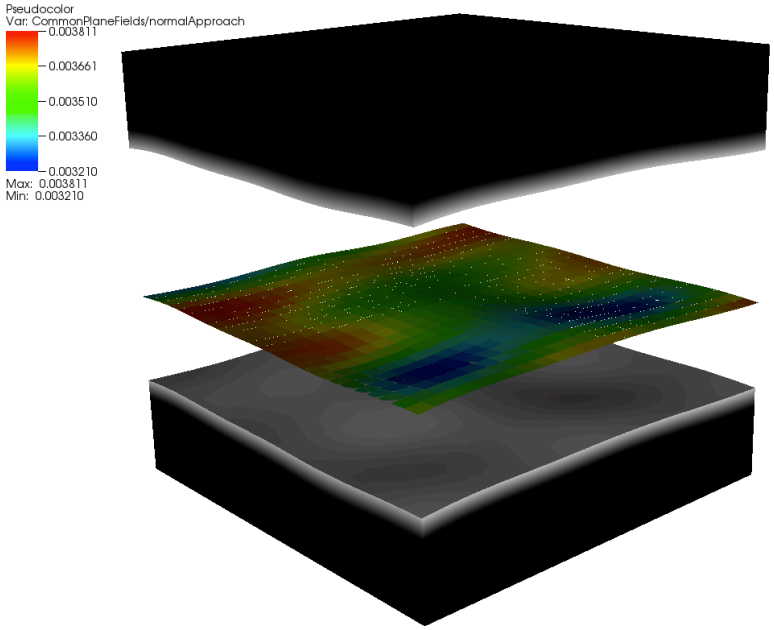


Figure 7: Illustration of the contact between two sliding interfaces from an initially mated fractal fracture surface

In addition to constitutive modeling, we have also been engaged in basic research on the numerical methods to resolve small-displacement contact implicitly. A stabilized algorithm based on Nitsche's method has been developed to enforce contact constraints over continuous and discontinuous surfaces in the finite element method.

This displacement-based method is more computationally efficient than penalty methods and Lagrange multiplier methods, and it results in well-conditioned system matrices. This allows us to now evaluate contacts implicitly in 2D with both simple Coulombic frictional response and frictionless response; significant progress has also been made in applying the same techniques in 3D.

We will be extending this to address more complicated surface interaction behaviors (e.g., Barton-Bandis, etc.).

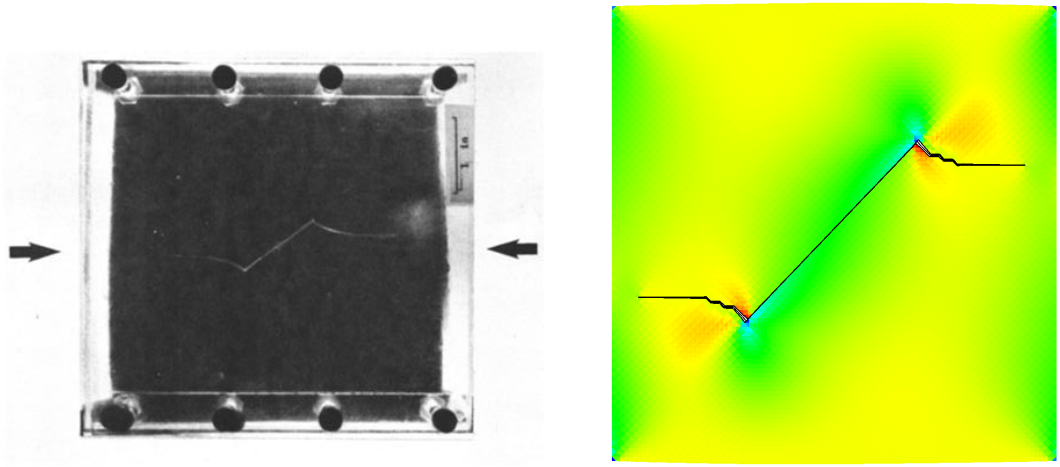


Figure 8: Implicit contact algorithm in 2D applied to sample with a pre-existing diagonal fracture (right) compared with experiment (left).

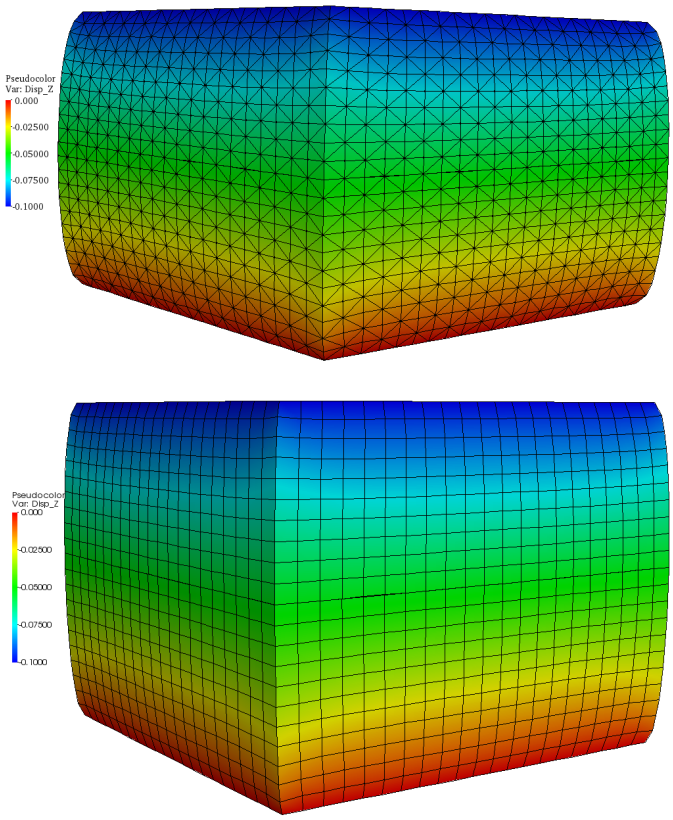


Figure 9: z-displacement contours for the 3D horizontal crack under frictionless sliding using Nitsche's method with tetrahedral (top) and hexahedral elements (bottom)

3.2.2 Proppant Modeling

The flow model includes several different fluid rheologies. These are currently fixed in the flow solver; however, future developments would move these into a fluid rheology library, allowing a user to select the model appropriate for the analysis. This would also allow the user to introduce new fluids without directly modifying each of the flow solvers.

The proppant transport module contains a fixed number of different options for describing settling and transport of the proppant particles. Future development would allow the user the option of introducing additional forcing terms, including wall effects and the effect of friction on the particle slip velocity.

Currently the proppant is assumed to consist of mono-disperse particles. Future development would extend the proppant transport module to account for separate populations of proppant types and separate distributions of proppant particles.

In order to maintain a sharp boundary between the settled and free flowing particles, the proppant is modeled as both a body of particles in suspension and an established proppant pack. The proppant pack currently behaves as a rigid body with respect to the mechanics solver – maintaining the same aperture regardless of

the stress experienced by the fracture. Future development would allow the pack to respond to changes in stress, decreasing porosity with increased pressure.

3.2.3 Flow and Transport Solver

We have completed the initial development for a three-dimensional finite element implementation of a flow and transport solver for dual permeability systems by leveraging previous code development experience and expertise from the Nonisothermal, Unsaturated-Saturated Flow and Transport Model (NUFT) software project.

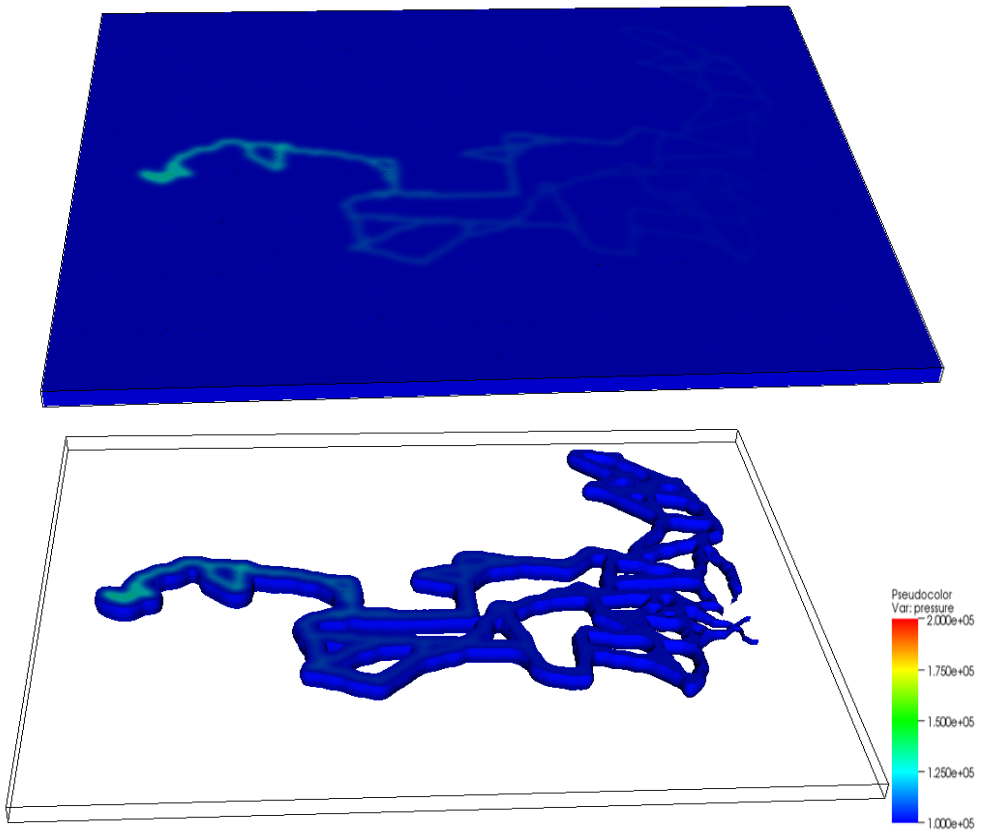


Figure 10: Transient pressure behaviour (Rock permeability = 0.01 mD)

The summary of this development is:

- A control volume finite element (FE) based discrete fracture model has been developed within the GEOS framework.
- Compared to the previously implemented cell-centered finite volume model this finite element model is capable of more accurately handling very complex geological features and fracture geometries.
- The model is able to provide high-fidelity near field solutions for fracture flow and thermal (solute) transport, which can therefore be used to help perform up-scaling studies, and calibrate current reservoir-scale continuum models (e.g. equivalent continuum, dual-porosity/dual permeability models).
- The control volume finite element based discrete fracture flow model, which is implemented within the GEOS framework, has been further tested.

- A couple of new element types (“line” and “ triangle-shell”) have been added in order to handle 2D/3D coupled fluid flow in both discrete fractures and surrounding matrix rocks.
- A 3D discrete fracture flow test case has been developed.
- GEOS parallel computing capability has been tested by running large 3D fracture flow test cases with using a variety of numbers of processors, ranging from 16 to 256. We have also conducted a preliminary scalability analysis of GEOS parallel performance.

3.2.4 Wave Propagation Solver

We have completed the initial development for a three-dimensional finite difference based wave propagation solver leveraging previous code development experience and expertise from the WPP and SW4 software projects.

This enables the next phase of microseismic source term modeling by allowing for direct coupling of the source term into ground motion solutions using a consistent reservoir representation.

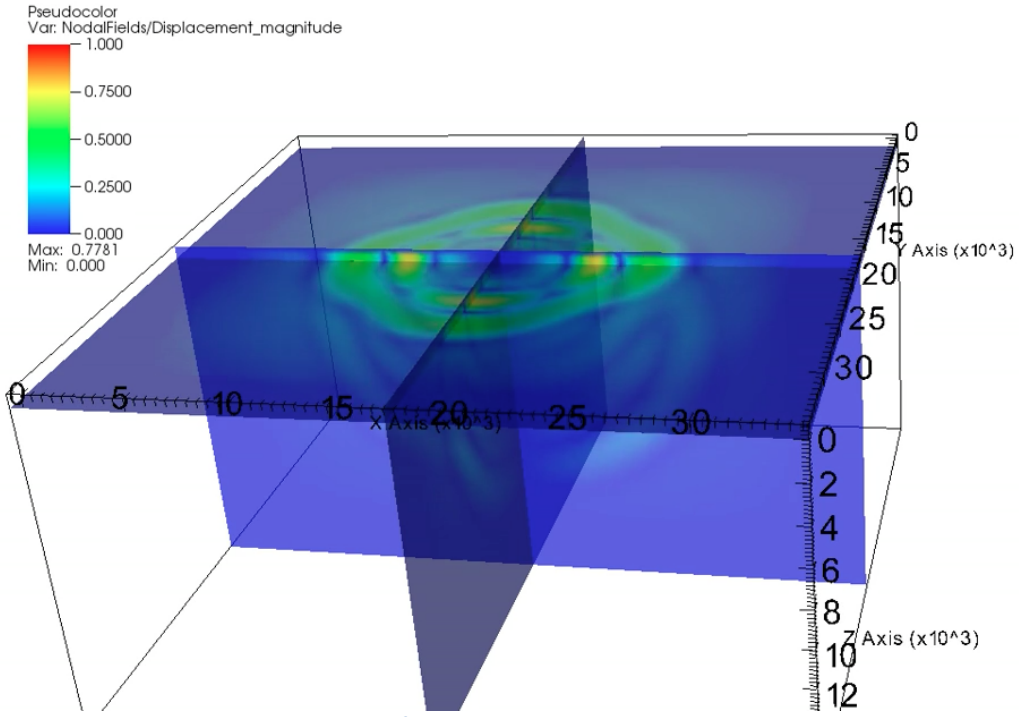


Figure 11: Point source propagation calculated from GEOS; results are identical to SW4

3.2.5 Material Modeling

We have also focused on supporting a wider range of material behaviors. To accomplish this goal we are (1) implementing an interface to the GEODYN material library and (2) developing a new materials model.

The GEODYN material library provides access to previously developed anisotropic damage models, which had been developed for granite. We have finished and cross code validated the implementation of the GEODYN material model.

We have also embarked on a new effort this quarter, which first identified a number of candidate models for shales under both quasi-static and dynamic loading. We then selected to focus on one such model by Crook, which exhibits both anisotropic elasticity and anisotropic plasticity. This model is an extension of work by Cazacu and Pietruszczak to extend the isotopic modified Cam Clay critical state model. We have developed the first iteration of an implementation of this model.

3.2.6 Other Developments

We have also focused on supporting a wider range of reservoir situations. In this quarter, we have implemented additional options for boundary and initial conditions.

We implemented the ability to specify both material properties and initial stress states according to three-dimensional tables of scalars, vectors, or tensors (depending on the state to specify) or by analytical functions for scalar properties. This allows us to now input realistic spatial distributions of properties, including bulk modulus, stress, density, etc.

4. Future Work

Fracturing during hydraulic stimulation is most directly and immediately observed by the microseismic response. GEOS simulates the time history of hydraulically induced fracture, and comparison with observed seismicity requires that GEOS generate a source term and then pass that to a seismic wave propagation code to produce synthetic seismograms for surface and borehole seismic arrays. This is one of the initiative's primary deliverables and is the target deliverable for the final phase of the project.

1. Couple finite difference solver with the microseismic source generation term
    a. Enabled by recent completion of an initial version of the finite difference wave propagation solver
    b. Will provide a tool that can input a pumping schedule and output ground motions.

2. Bi-directionally couple microseismic source term model with constitutive model
    a. Will be enabled by the completion of the anisotropic elasto-plastic model based on an extension of the Cam Clay model
    b. Will provide a realistic mechanical behavior changes due to the generation of seismicity-inducing damage.

3. Tri-directionally couple microseismic source term model with constitutive model

a. Enabled by recent completion of an initial version of the finite element based flow and transport solver
b. Will provide a realistic hydro-mechanical behavior changes due to the generation of seismicity-inducing damage.

4. Multi-scale refinement
    a. Dual-scale paradigm for sequential, hierarchical multi-scale treatment
    b. Will use Continuum Damage Mechanics (CDM) to predict the statistical average response of cracked solids, without describing the real geometry of each micro-crack
    c. Will provide more realistic representation of the coupled hydro-mechanical representation across multiple scales.

4.1 Multi-scale Damage Modeling

For the last phase, it is necessary to pursue such a strategy in order to reduce the number of degrees-of-freedom in the problem to a computationally tractable size given that the memory and processing capacities of the HPC resources are already anticipated to be limiting for the applications of interest. In micro-mechanical damage models, the main challenge consists of characterizing the set of cracks present in the medium according to their size and orientation. Within each set, crack growth is generally controlled by a Griffith criterion. The natural dissipation variable should be the length of the cracks within the set considered, but crack densities are generally preferred. For each crack set, the thermodynamic variable conjugate to crack density is referred to as the “affinity” or “damage force”. The Griffith criterion is met when the derivative of the affinity to crack density is more than a certain critical value. If the threshold is reached, crack densities must be updated. The damaged stiffness tensor can then be constructed by accounting for the updated crack geometry in an appropriate homogenization scheme (e.g. the self-consistent method or Mori-Tanaka scheme).

In general terms, the steps necessary to implement this will be:

1. Determine constitutive relationships to predict microscopic crack initiation and propagation.
    a. Characterize the microstructure parameters and their conjugate affinities.
    b. Relate affinities to stress and microstructure parameters to texture-related functions.
2. Determine the functional dependence of RVE size on microstructural properties.
3. Determine appropriate coupling terms to achieve equivalence between continuum (stochastic) and discrete representations.
4. Determine thresholds at which to advect microstructural damage into explicitly represented fracture surfaces.

Many parts of this approach are well-defined; however, despite some research, the

connection between microstructural parameters and their conjugate affinities will be an area that requires research to properly address. In general, the completion timeline for the multi-scale approach will be difficult to forecast in the absence of identified funding.

4.2 Uncertainty Quantification

Evaluating the uncertainty is an essential element of parametric testing of GEOS, its application to forward predictions of reservoir stimulation. Also, by constraining the sensitivity of various predictions to variations in reservoir and operational variables, UQ constitutes an important precursory step to the development of derivative, reduced complexity, fast running tools. Along with operational parameters, there are a variety of uncertainties associated with the geologic model including the nature of pre-existing fracture distributions and rock mechanical properties which require evaluation if optimization is to be achieved.

We have linked GEOS with the LLNL UQ code, PSUADE (Problem Solving environment for Uncertainty quantification And Design Exploration) which is a suite of uncertainty quantification modules capable of addressing high-dimensional sampling, parameter screening, global sensitivity analysis, response surface analysis, uncertainty assessment, numerical calibration, and optimization. Sensitivity curves have been established for a number of process and geologic parameters in two- and three-dimensional systems; however, a number of studies remain to be performed.

4.3 High Strain Rate Fracturing

While much of our current effort is directed at optimization of the hydraulic stimulation, we recognize the need for a computational tool to aid in the design of new stimulation methods that minimize water use and potentially involve the use of energetic materials. We have investigated the application of some types of energetic materials to the subsurface, using the GEOS tool; however, a number of applications remain beyond our capabilities.

Here we describe the requirements for three separate scenarios utilizing energetic stimulation. In the simplest case, we could use the current infrastructure with a special time-dependent joint model to capture the behavior. In the ideal case, full coupling of GEOS with GEODYN, a parallel transient Eulerian finite volume code with constitutive models that are well suited for analyzing the dynamic response of geologic media, would be undertaken. Use cases include:

4.3.1 Deflagration-Induced Pressure Elevation in Fracture Network

In this case we do not need to consider the propagation of the shockwave associated with detonation, but rather the introduction deflagration products as a source term to produce the higher pressures. This is relatively straightforward within the existing GEOS framework, and has been accomplished and demonstrated via a special boundary condition as part of the initiative.

4.3.2 Detonation-Induced Pressure Pulse in Fracture Network

Simulation of this scenario, would require modification of our basic hydraulic fracturing capability to enable:

1. Transmission of waves through the fluid.
2. Production of the actual shock.
3. Inelastic material models

The confined nature of this problem may or may not lend itself to a simple quasi-two-dimensional empirical approach that takes advantage of the parallel plate geometry of a fracture. This application has been nominally investigated but requires further study to determine whether GEOS should be extended in this way.

4.3.3 Shock Wave Interaction Induced Tensile Damage Zones

This type of simulation requires strong coupling between GEOS and GEODYN. A fully coupled 3D GEOS/GEODYN code requires passing the fracture geometry to GEODYN which then interprets the geometry and constructs volume fractions on its internal regular grid representation, performs its calculations, then passes back vector forces for each face it was given. Another complication is that not only is re-gridding required but also information on the parallel partitioning of GEOS must be parceled out (efficiently) to the spatial partitions in GEODYN that require the information. We have addressed this with the Bifroest framework. Accomplishing this along with evaluation of alternative approaches (use GEODYN as an initializer to a Lagrangian code rather than a fully coupled simulation) would likely require at least 2 years of development effort.

Appendix A: Work-flow

GEOS uses the common simulation design pattern of: mesh generation, input file design, process, post-process. Specifically, the steps involved in an analysis include:

1. Design the "input deck", a XML formatted file that describes the problem you want to run and adheres to the XML Schema Definition (XSD) for GEOS.
2. Execute GEOS with the input deck file as an argument using the “-i" flag. In a cluster environment, this often requires submitting a “job” to a “batch manager”. The “batch manager” is software that is responsible for taking the computational job and the requested resources (e.g., 10k processors for 12 hours) and then either matches it with currently available resources or queues the job according to the priority of the “bank” you are using. A bank has a priority commensurate with the amount of time purchased with that bank adjusted by the usage (for Livermore Computing, LC, this usage metric is calculated over a two-week window).
3. Once the job has executed (and often mid-execution to make sure everything is running smoothly), the results can be visualized and queried. the VisIt parallel visualization software (downloadable from http://visit.llnl.gov) to render and query the results for the desired data. This has been optimized for the LC systems.

We usually work on assembling workflows before running any analysis that will need to be run multiple times with variations between runs. The workflows often consist of Python or shell scripts to accomplish the steps above. This can mean extra time to set everything up, but it ensures that each run is interpreted using the same protocol and that the results remain comparable.

Appendix B: Quality Assurance

Our development strategy involves having “development” and "production" lines of the code. The production version represents the stable version of the code. Unless otherwise noted, this is the version used to provide results to external parties.

We also have a number of parallel “development” branches, which represent different capabilities of differing maturity levels, which are under active development and may not necessarily be ready to provide reliable results. The implicit solver is being developed in one such branch of the code as are the implementations of the dual permeability flow and transport and wave propagation. Once a capability is robust and reintegrated with the production source, it is made available for performing analyses for external parties.

When a capability is migrated from development to production, it is typically treated as an addition to the set of capabilities. For instance, once the implicit solver is fully tested and migrated to production, a user would be able to run either with the current dynamic relaxation approach or with the fully implicit solver. In order for such a development branch to be integrated into the production line, however, it must first pass a test suite (currently, 400 tests), which is meant to ensure that the results from previous versions of the production line remain unchanged with changes to the source code. Currently, the standard is high, requiring that results generated with successive versions of the code must be the same down to a binary comparison of files produced (for those tests common to both versions). In addition to passing the test suite, we also advise the developer to include a new test (or tests) to ensure that any future development will not compromise the capabilities he/she implemented.

This represents a fairly common software development and quality assurance paradigm. Because of this, we are able to use a number of powerful tools developed, used, and supported by others to help us manage the process. We currently use the ATS test suite management system (developed internally for the parallel environment at LLNL) and a software version control system (the open source Git) to reduce the effort (and possible sources of error) in managing the development and integration process.

Appendix C: Publications

PROCEEDINGS, Thirty-Seventh Workshop on Geothermal Reservoir Engineering
Stanford University, Stanford, California, January 30 - February 1, 2012
SGP-TR-194

SIMULATION OF HYDRAULIC FRACTURE NETWORKS IN THREE DIMENSIONS

Randolph Settgast, Scott Johnson, Pengcheng Fu, Stuart D.C. Walsh, Frederick Ryerson

Atmospheric, Earth, and Energy Division, Lawrence Livermore National Laboratory
7000 East Ave., L-286
Livermore, CA 94550, USA
e-mail: settgast1@llnl.gov

ABSTRACT

Hydraulic fracturing has been an enabling technology for commercially stimulating fracture networks for over half of a century. It has become one of the most widespread technologies for engineering subsurface fracture systems. Despite the ubiquity of this technique in the field, understanding and prediction of the hydraulic induced propagation of the fracture network in realistic, heterogeneous reservoirs has been limited. Recent developments allowing the modeling of complex fracture propagation and advances in quantifying solution uncertainties, provide the possibility of capturing both the fracturing behavior and longer term permeability evolution of rock masses under hydraulic loading across both dynamic and viscosity dominated regimes. We present a framework for leveraging these advances in practical workflows for analyzing prospective and operating geothermal / hydrothermal sites. We will demonstrate the first phase of this effort through illustrations of fully three-dimensional, 2-way coupled hydromechanical simulations of hydraulically induced fracture network propagation and discuss preliminary results regarding the mechanisms by which fracture interactions and the accompanying changes to the stress field can lead to deleterious or beneficial changes to the fracture network.

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

INTRODUCTION

Reservoir stimulation through hydraulic fracture is a critical element in unlocking the potential of geothermal energy production where natural permeability is inadequate (Figure 1). However, despite the fact that hydraulic fracturing has been in employed for almost 60 years, having first been used to simulate oil and gas wells in the early 1950's, many uncertainties still surround its use. In particular, given the public scrutiny of hydraulic fracture operations due to concerns regarding its environmental impact and potential for induced seismicity, there is a need for robust models capable of determining:

- The impact of hydraulic fracturing on caprock integrity;
- Whether fractures will propagate as designed;
- What seismicity will be induced as a result of the fracturing operation;
- Whether recovered fluids will be released; and importantly,
- Whether production rates will meet expectations.

Addressing these questions is a difficult task. Modeling hydraulic fracturing in the presence of a natural fracture network is a complicated multi-physics, multi-scale problem due to the coupling between fluid, rock matrix, and rock joints, as well as the interactions between propagating new fractures and existing natural fractures. Nevertheless, in recent years, a number of advances have allowed researchers in related fields to tackle the modeling of complex fracture propagation as well as the mechanics of heterogeneous systems. These developments, combined with advances in quantifying solution uncertainties, provide possibilities for the geologic modeling community to capture both the fracturing behavior and longer term permeability evolution of rock masses under hydraulic loading across both dynamic and viscosity dominated regimes.

This paper describes the development of a computational capability focused on the creation, characterization, maintenance, and active management of optimal fracture networks for energy extraction from enhanced geothermal systems. The primary component of this capability is the development of a high-fidelity geomechanics code, GEOS, a multi-scale, multi-physics, fracture mechanics model that will describe the development

of fracture networks for different lithologies and applications as a function of initial geologic conditions, regional stress and stimulation work flows. Here we present the first phase of this effort through illustrations of fully three-dimensional, 2-way coupled hydromechanical simulations of hydraulically induced fracture network propagation and discuss preliminary results regarding the mechanisms by which fracture interactions and the accompanying changes to the stress field can lead to deleterious or beneficial changes to the fracture network.

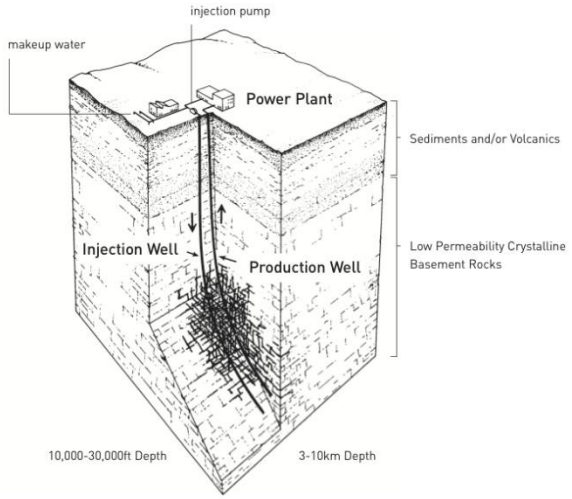


Figure 1: Schematic of an Enhanced Geothermal System with injection and production wells extracting heat from a stimulated volume of rock with low initial permeability [1].

MECHANICS AND FLOW

Mechanics

In this work, the deformation of meshed volumes is governed by a series of standard Lagrangian FEM solvers. Fundamentally these solvers enforce some form of the equations of motion

$$\nabla \cdot \mathbf{T} + \rho(\mathbf{b} - \mathbf{a}) = \mathbf{0}. \tag{1}$$

Depending on the application, (1) can take the form of a general mechanics solution, a static solution (**a**=**0**), as well as being cast in terms of first or second Piola-Kirchhoff tensors. In the cases presented here, an explicit dynamics solver is applied to the first Piola-Kirchhoff forms of (1).

Parallel Plate Flow

The flow of the fluid through the fractures is assumed adhere to the parallel plate flow assumptions [2,3]. Given a single edge connected to *n* faces where each face is given a local index *i*. To solve for the flow, we calculate the mass flux between the edge and each face. The fracture permeability of between the face and the edge κ_i is given as

$$\kappa_i = \frac{\alpha_i^3 w}{12 \mu L_i}, \tag{1}$$

where α_i is the hydraulic aperture of the face, w is the length of the edge, μ is the dynamic viscosity, and L_i is the length from the center of the face to the center of the edge. The mass flow rate ($\dot{m}_i$) from the face to the edge is easily expressed as

$$\dot{m}_i = \kappa_i (\rho_i P_i - \rho_e P_e), \tag{2}$$

where ρ_i is the density of the face, P_i is the fluid pressure on the face, ρ_e is the density at the edge, and P_e is the pressure at the edge. Applying conservation of mass at the edge provides a solution of $\rho_e P_e$ as

$$\rho_e P_e = \frac{\sum_i^n \kappa_i \rho_i P_i}{\sum_i^n \kappa_i}. \tag{3}$$

Substituting (3) into (2) allows for the solution of the mass flux between the edge and the face. These relations (1-3) are implemented in both a time explicit transient solver, and a steady-state implicit solver.

FRACTURE

The focus of this work to date is to provide the topological flexibility to model the creation of new surfaces. To this end, tools for splitting a mesh similar to those described by Settgast[4] have been further developed in the GEOS framework. In this approach, nodes, edges, and faces are split along element boundaries into separate entities. This is achieved when a closed path of faces that have attained a ruptured state can be found. This method of splitting along element boundaries is performed on a node-by-node basis and can be described through the following list of procedures:

1. Determine the state on faces, and mark the faces that have achieved a ruptured state.
2. For each node, find a closed path of faces that make up a "rupture plane" about which the node can be split.
3. Split the node, any edges and faces.
4. Repair connectivity between the nodes, edges, faces, and elements.

With the preceding, the application of computational fracture mechanics is feasible. There are many methods by which to attempt this ranging from various cohesive zone approaches [4-6]. In this study

we do not yet apply any of these methods, instead the mesh is simply broken, extending the fracture in the process. The implementation of a method for smooth crack opening will be pursued in future work.

In the case of field scale studies, the mesh resolution required to resolve the stress field near the tip of a fracture is unattainable without some form of automatic mesh refinement near the tip. In cases where this is the case, methods may be used to estimate the stress field as given in [7]. If no such method is utilized, then a stress criteria for face rupture will likely be a compressive stress, as the unbounded stress field is dramatically under-estimated by the resolution.

SURFACE CONTACT

Once fracture surfaces have been generated, they must be prevented from subsequent inter-penetration. In the general case, a contact methodology that allows for shear slip along the surfaces is desired. The general contact enforcement method in GEOS is a variation of the so-called "common-plane" (CP) method. The CP method institutes face-to-face contact by detecting overlapping face geometries, and producing a penalty force resisting the contact. This approach is essentially described in described in [8], although refinements and modifications have been made.

An alternative to the generalized approach, which bears significant computational cost, a simple method of surface contact enforcement is available when no shear slip is present. In this case, a penalty stiffness is enforced on the inter-penetration distance of formerly coincident faces (i.e. faces that used to be the same face prior to rupture). This method is used in the work presented in this study.

COUPLED MECHANICS/FRACTURE/FLOW

The coupling of mechanics solver with fracture capability to a fluid solver to represent the flow through the fractures is relatively straightforward. The first step is to define the flow mesh once the volumetric mesh is split. While there are many options for this, the method presented here is to define the flow mesh on the original faces/edges/nodes that have been split. While these objects are likely to be disconnected from each other, an alternate connectivity that links the original edges to the original faces (recall that these relations have been changed by the fracture process) can be used to define a contiguous mesh as far as the flow solver is concerned.

Once a flow mesh is defined we focus on the method to couple the two solvers together. In essence the procedure for a time-explicit coupled solver is as follows:

1. Perform flow solve using beginning of step apertures, and fluid pressures. This gives the mass in each fluid volume at the end of the step.
$$FluidSolve(P^n, \alpha^n) \Rightarrow m^{n+1}$$
2. Update nodal velocities to mid-step, and displacements to end-of-step.
$$\mathbf{v}^{n+1/2} = \mathbf{v}^{n-1/2} + \mathbf{a}^n \Delta t^n$$
$$\mathbf{u}^{n+1} = \mathbf{u}^n + \mathbf{v}^{n+1/2} \Delta t^{n+1/2}$$
3. Update material state of the solid volume, and generate nodal forces from those volumes.
$$MatUpdt\left(\frac{\partial \mathbf{v}^{n+1/2}}{\partial \mathbf{x}^{n+1/2}}, Q^n, \Delta t^{n+1/2}\right) \Rightarrow Q^{n^{n+1}}$$
4. Update the fluid pressure using the mass from step 1, and the volume at the end of the step. Specifically volume is the gap between the physical faces multiplied by the face area.
$$EOS\left(m^{n+1}, V^{n+1}\right) \Rightarrow P^{n^{n+1}}$$
5. Apply fluid pressure from a flow face as a boundary condition on the physical faces that are related to it. This is a simple pressure boundary condition to the mechanics solver.
6. Calculate the acceleration of the nodes at the end of the step.

EXAMPLES

In the following examples, a linear elastic material is used for solid materials, and a simple linear equation of state is used for the fluid. The material properties are summarized in Table 1.

Table 1: Summary of Material properties.

Property	Value
Solid Bulk Modulus	15.0 GPa
Solid Shear Modulus	15.0 GPa
Fluid Bulk Modulus	0.1 GPa
Fluid Pressure	7.0 MPa
Min Horizontal Confinement	6.0 MPa
Max Horizontal Confinement	8.0 MPa
Vertical Confinement	10.0 MPa
Dynamic Viscosity (k)	1.0e-3 N s/m^2
Rupture Stress Criteria*	5.2 MPa

* note that the compressive value of rupture stress criteria is due to the under-resolution of the mesh, as discussed in the preceding section of fracture.

The first example is a pseudo-2d representation of a horizontal plane (200m x 200m) with a 3 pre-existing

fractures. The middle fracture runs perpendicular to the minimum principle stress, while the end fractures run in the direction of maximum horizontal principle stress as shown in Figure 1. The middle fracture is then pressured at its center, and a "tensile" stress develops at the crack tip as shown in Figure 1b. As shown in the middle figure, the fracture propagates until it joins with the end fractures. At this point the end fractures are pressurized (Figure 1c) and the simulation is terminated.

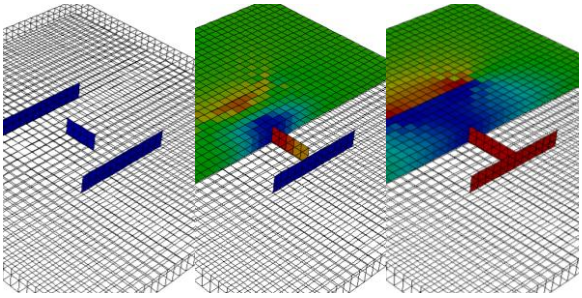


Figure 2: Hydraulically induced extension of a pre-existing fracture terminating on a perpendicular fault using on a pseudo-2d problem. Color scale indicates pressure, and stress in the minimum horizontal direction.

The next example is the 3-dimensional hydraulic fracture propagation in a (200m x 200m x 200m) block. As was the case in the last example, a single fracture runs in the direction of minimum horizontal stress. In this case however, only a single fracture exists in the direction maximum horizontal stress. When the fracture is pressurized, the fracture adopts a circular shape, and begins to extend maintaining its shape. As expected, when the growing fracture joins with a pre-existing perpendicular fracture, growth ceases, and does not restart until both fractures are pressurized – at which point they begin to grow together.

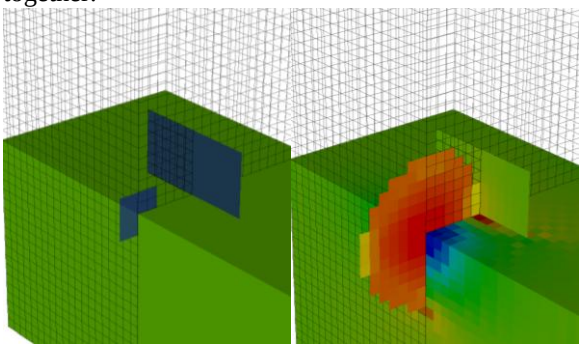


Figure 3: Hydraulically induced extension of a pre-existing fracture terminating on a perpendicular fault using on a 3-dimensional problem. Color scale indicates pressure, and stress in the minimum horizontal direction.

While the preceding examples are a good illustration of rudimentary capabilities and allow for understanding the mechanisms at play during fracture propagation, our end goal is to model the stimulation of a large scale fracture network in 3-dimensions, as was done in 2-dimensions in [9]. While the capability for modeling this problem is remains under development, initial progress has been made. In the following example, a flow calculation is performed on the same 200m block with a pre-existing 3-dimensional joint set as shown in Figure 3. A well source is placed in the lower near corner, while a recovery well is placed in the upper far corner. The source well pressure is specified as 7 MPa, the recovery well pressure is specified as 5 MPa, and hydraulic aperture is fixed at 10 μm.

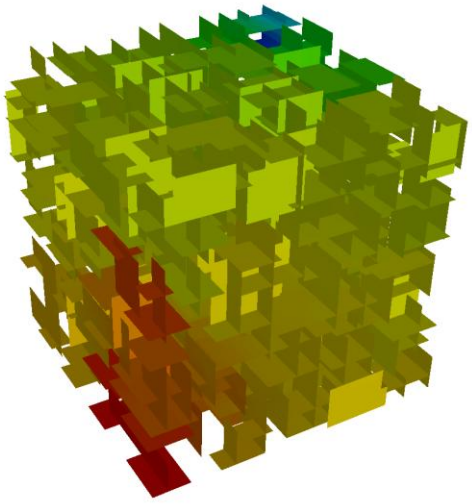


Figure 4: Flow calculation between a source and extraction well on block with a 3-dimensional joint set. Color scale indicates fluid pressure.

CONCLUSIONS AND DIRECTION OF FUTURE WORK

In this study a methodology to simulate the evolution of fracture networks in 3-dimensions through a 2-way coupled finite element approach has been presented. Simple examples of hydraulically driven fracture, including the ability to join fractures has been shown. In addition, a simulation of flow through a set of three-dimensional fractures has been shown. While the basic capabilities are promising, additional capabilities are required to achieve the goals of simulation of realistic fracture networks shown in Figure 5. To this end, future work seeks to develop and implement the following:

1. Implementation of quadratic tetrahedral elements for greater flexibility in fracture propagation direction.
2. Development and implementation of a method to estimate crack-tip stresses in 3-dimensions.
3. Implementation of an AMR capability to greater resolve the material states at the crack-tips.
4. A complete suite of implicit and explicit solvers to address different time scales, and the ability to transition between scales automatically.

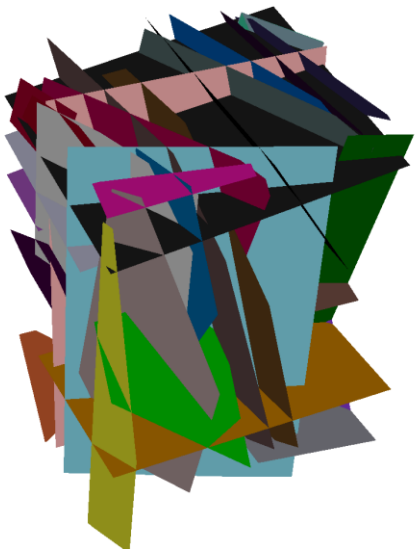


Figure 5: A realistic fault set derived from experimental data [10].

ACKNOWLEDGMENTS

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

REFERENCES

1. MIT (2006), "The Future of Geothermal Energy: Impact of Enhanced Geothermal Systems (EGS) on the United States in the 21st Century" Rep., 372 pp, Massachusetts Institute of Technology.
2. Adachi, J., Siebrits, E., Peirce, A., and Desroches, J. (2007) "Computer simulation of hydraulic fractures", *International Journal of Rock Mechanics and Mining Sciences*, **44**: 739-757.
3. Johnson, S.M., Morris, J.P. (2009) "Model Development to Characterize Hydraulic Fracturing for Geologic Carbon Sequestration Applications." *International Conference on Rock Joints and Jointed Rock Masses*, Tucson, AZ.
4. Settgast, R.R, Rashid, M.M, (2009) "Continuum coupled cohesive zone elements for analysis of fracture in solid bodies", *Engineering Fracture Mechanics*, **76**, p. 1614-1635.
5. Xu XP, Needleman A. (1995), "Numerical simulations of dynamic crack growth along interface". *Int J Fracture*, **74(4)**:289–324.
6. Sam Chin-Hang, Papoulia KD, Vavasis SA. (2005) "Obtaining initially rigid cohesive finite element models that are temporally convergent." *Engng Fract Mech*, 2005; **72**:2247–57.
7. Fu, P., Johnson, S.M., Settgast, R.R., and Carrigan, C.R. (2011). "Generalized displacement correlation method for estimating stress intensity factors." *Engineering Fracture Mechanics*, in review.
8. Vorobiev, O. (2011), "Simple Common Plane Algorithm", *International Journal of Numerical Methods in Engineering*, 10.1002/nme.3324.
9. Fu, P., Johnson, S.M., Hao, Y., and Carrigan, C.R. (2011) "Fully coupled geomechanics and discrete flow network modeling of hydraulic fracturing for geothermal applications." *The 36th Stanford Geothermal Workshop*, Jan. 31 – Feb. 2, 2011, Stanford, CA.
10. Lin, W;Blair, S C;Wilder, D;Carlson, S;Wagoner, J;DeLoach, L;Danko, G;Ramirez, A L; Lee, K. (2001), "Large Block Test Final Report", UCRL-ID-132246-REV-2. 442pp.

Engineering Fracture Mechanics 88 (2012) 90–107

Contents lists available at SciVerse ScienceDirect

Engineering Fracture Mechanics

journal homepage: www.elsevier.com/locate/engfracmech

Generalized displacement correlation method for estimating stress intensity factors

Pengcheng Fu *, Scott M. Johnson, Randolph R. Settgast, Charles R. Carrigan

Atmospheric, Earth, and Energy Division, Lawrence Livermore National Laboratory, 7000 East Avenue, L-286, Livermore, CA 94550, United States

ARTICLE INFO

Article history:
Received 1 March 2012
Accepted 8 April 2012

Keywords:
Fracture mechanics
Stress intensity factor
Displacement correlation method
Quarter-point element
Fracture propagation
Fracture interaction

ABSTRACT

Conventional displacement-based methods for estimating stress intensity factors require special quarter-point finite elements in the first layer of elements around the fracture tip and substantial near-tip region mesh refinement. This paper presents a generalized form of the displacement correlation method (the GDC method), which can use any linear or quadratic finite element type with homogeneous meshing without local refinement. These two features are critical for modeling dynamic fracture propagation problems where locations of fractures are not known *a priori*. Because regular finite elements' shape functions do not include the square-root terms, which are required for accurately representing the near-tip displacement field, the GDC method is enriched via a correction multiplier term. This paper develops the formulation of the GDC method and includes a number of numerical examples, especially those consisting of multiple interacting fractures. We find that the proposed method using quadratic elements is accurate for mode-I and mode-II fracturing, including for very coarse meshes. An alternative formulation using linear elements is also demonstrated to be accurate for mode-I fracturing, and acceptable mode-II results for most engineering applications can be obtained with appropriate mesh resolution, which remains considerably less than that required by most other methods for estimating stress intensities.

1. Introduction

The stress intensity factor (SIF) is an important concept in fracture mechanics for relating stress and energy release rate at the fracture tip to loading and crack geometry. Although closed-form analytical solutions are available for a number of special fracture-load configurations (many of which have been compiled in [1]), SIF's are often calculated in the context of numerical methods, especially the finite element method (FEM) for arbitrary fracture-load configurations. Several methods are available for calculating or estimating SIF's with the FEM, such as the *J*-integral [2] and its variants, the stiffness derivative technique [3], and a suite of methods based on the interpretation of near-tip nodal displacements. In the last category, there are at least three variants, including the displacement extrapolation method [4–7], the quarter-point displacement method [8], and the displacement correlation method [9,10]. These methods and others have been compared in a number of studies [e.g. 5,11–14]. One of the most significant advantages of the displacement-based methods is the simple formulation. Although the displacement-based methods were often found to be less accurate than the *J*-integral or the stiffness derivative method under certain conditions, the accuracy remains acceptable for most engineering applications [e.g. 5,14]. Many of the displacement-based methods were developed in the 1970s and 1980s in tandem with various special "quarter-point" finite element types [15–19] and transition elements [20] used in these methods. Though few new developments have been reported on the displacement-based methods in the intervening decades [21], they continue to be widely used.

* Corresponding author. Tel.: +1 925 422 3579.
E-mail address: fu4@llnl.gov (P. Fu).

0013-7944/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.engfracmech.2012.04.010

Nomenclature

a, b, c, h	variables denoting key geometrical characteristics of fracture-solid systems
$C_I^A, C_{II}^A, C_I^B, C_{II}^B$	correction multipliers for the GDC method. The superscript denotes whether it is for Method A or Method B; the subscript indicates the mode of fracture
$f_r^a, f_r^b, f_r^c, f_\theta^a, f_\theta^b, f_\theta^c$	functions of the angular coordinate of a point to simplify the expression of certain equations
F_I, F_{II}	fracture-configuration correction factors related to geometrical characteristics of the fracture-solid system
G	shear modulus of the solid
K_I, K_{II}	mode-I and mode-II stress intensity factors
l_E	characteristic length of a finite element
P	point load applied on a bending beam with a mid-span notch
r, θ	polar coordinates of a point
s	span width of a bending beam with a mid-span notch
$u_r^a, u_r^b, u_r^c, u_\theta^a, u_\theta^b, u_\theta^c$	radial and angular displacements of a point due to the three boundary condition modes (a, b, and c de noted in Fig. 1)
u_r, u_θ	radial and angular displacements of a reference point. A superscript might be used to denote the reference point
β	material constant depending on whether this is a plain-stress or plain-strain problem
δ	mesh perturbation factor
ν	Poisson's ratio of the solid
$\sigma_f, \tau_f, \sigma_x, \sigma_y$	far-field stress components of the fracture-solid system
σ_c, τ_c	fracture surface stress components
GDC	generalized displacement correlation (method)
SIF	stress intensity factor

The J-integral as well as its variants (e.g. M-integral) and the displacement-based methods all require accurate resolution of near-tip displacement/stress/strain fields. Therefore special element types (e.g. quarter-point elements) and/or near-tip element refinement are usually used as accuracy enhancement/assurance measures. The present study is motivated by engineering applications where such enhancement is impractical in terms of computational cost but a moderate error margin (e.g. 10%) is acceptable. The simulation of hydraulic fracturing in natural fracture systems represents this class of applications and is the direct motivation for the current study [22]. Simulating a hydraulic fracturing process usually involves modeling multiple cracks propagating along arbitrary paths, so the locations of the crack tips are not specified *a priori*. Employing any near-tip enhancement measures necessitates heterogeneous meshing, which is theoretically possible to handle but requires complex variable mapping between meshes every time a fracture advances. This is extremely costly for hydraulic fracturing simulations because very small time steps have to be adopted to handle the wide spectrum of length-scales that have to be resolved, spanning from tens of micrometers (the aperture width of typical rock joints) to hundreds of meters (dimension of the reservoir). Therefore, homogeneous and relatively coarse meshing without local refinement or frequent remeshing and variable mapping is the only viable option. Additionally, because we have to frequently handle the situation where two fractures are close to each other before they intersect, it is desired to only use information in the first layer of element surrounding a fracture tip. On the other hand, owing to the inherently high variability in rock properties and the high uncertainty in the determination of rock properties, an error of 10–20% in the estimation of SIF is considered well acceptable.

The goal of this study is to develop and verify a displacement-based method, termed the generalized displacement correlation (GDC) method for estimating SIF, which uses regular finite element types and does not require local mesh refinement. In the currently paper, we first review the mechanical and mathematical principles behind the original displacement-based methods in a generalized context in Section 2. Compared with the original derivation of these methods, the loading condition is generalized by including crack surface traction and the meshing scheme is generalized by circumventing the dependency on the specific shape functions of quarter-point elements. This new GDC formulation encompasses the original formulation based on quarter-point elements as a special case. Subsequently, we develop the new generalized formulation in Section 3 and further enhance its accuracy in Section 4 by introducing an empirical correction multiplier term. In Section 5, we test the new method against a number of fracture-load configurations with an emphasis on cases with inter-crack interactions, a situation critical to our hydraulic fracturing simulator development effort. The numerical examples in Sections 4 and 5 use the same Poisson's ratio and tip-region mesh configuration and use meshes based on a regular grid. The sensitivity of the results to the Poisson's ratio, near-tip mesh configurations, and mesh perturbation are evaluated in Sections 6 and 7.

2. Review of displacement-based methods in a generalized framework

Consider the two-dimensional (2D) continuum (linearly elastic and isotropic) around a crack tip as shown in Fig. 1, with far-field normal (σ_f) and shear (τ_f) stress existing along with crack surface traction (σ_c and τ_c). Note that "traction" in this

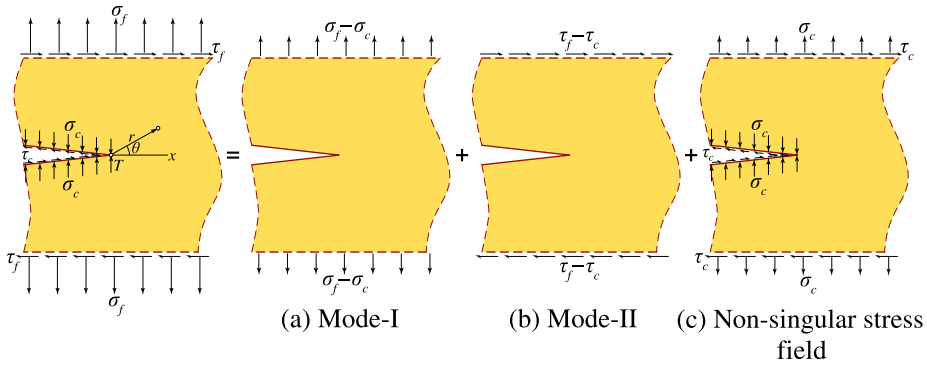


Fig. 1. The near-tip region of a 2D medium and the decomposition of fracture modes according to the superposition principle. The polar coordinate system used in this study is denoted in the figure. Fracture openings in this and other examples are exaggerated for illustration purposes.

paper refers to stress distributed along fracture surface while the same term is often used in cohesive zone models for a different meaning. Stresses σ_f, τ_f, σ_c, and τ_c are independent of each other, but the spatial variation of each of them is not considered. Their values can be either positive or negative, with the arrows in Fig. 1 indicating the positive stress directions. According to the superposition principle, the mechanical response of this system is the sum of the responses of the three cases [(a)–(c)] to the right of the equal sign in the figure. Case (a) and case (b) respectively correspond to the classical boundary/loading conditions for mode-I and mode-II fracturing, whereas in case (c) the crack surface traction balances the far-field stress. Only the stress conditions in the two former cases [(a) and (b)] induce stress/strain singularities in the near-tip region, while the latter case (c) generates homogeneous stress and displacement fields which contribute to the overall mechanical response but not the near-tip stress singularity. The loading conditions in case (a) and case (b) are the symmetric and skew-symmetric (antisymmetric) parts of the load that induce a near-tip stress singularity, respectively. Much of the development of *fracture mechanics* disregards the tractions along the crack surface, so case (a) and case (b) have been the focus of previous studies. In certain applications such as hydraulic fracturing, the pressure inside the fractures is the main mechanism for driving fracture extension with $\sigma_c < \sigma_f < 0$. Under such conditions, the stress condition in case (c) significantly contributes to the mechanical responses of the system and cannot be overlooked.

With higher-order terms omitted, the displacement field (relative to the crack tip displacement) induced by loading case (a) is

$$\begin{Bmatrix} u_r^a \\ u_\theta^a \end{Bmatrix} = \frac{K_I}{G}\sqrt{\frac{r}{2\pi}}\begin{Bmatrix} \cos\frac{\theta}{2} \\ -\sin\frac{\theta}{2} \end{Bmatrix}\left(\beta - \cos^2\frac{\theta}{2}\right) \tag{1}$$

where K_I is the mode-I stress intensity factor; G is the shear modulus of the medium; β is a constant depending on whether this is a plane strain ($\beta = 2[1-\nu]$ with ν being the Poisson's ratio) or a plane stress ($\beta = 2/[1+\nu]$) problem. It we assume that the elasticity parameters (G and β) are constants for a given problem, the equation can be simplified as

$$\begin{Bmatrix} u_r^a \\ u_\theta^a \end{Bmatrix} = K_I\sqrt{r}\begin{Bmatrix} f_r^a(\theta) \\ f_\theta^a(\theta) \end{Bmatrix} \tag{2}$$

where $f_r^a(\theta)$ and $f_\theta^a(\theta)$ are functions of the angular coordinate (θ) of the point where the displacement is measured. The effects of the elasticity parameters are incorporated into these two functions and they are considered constants for the purpose of this section. We can also write the corresponding equations for case (b), namely mode-II fracturing as

$$\begin{Bmatrix} u_r^b \\ u_\theta^b \end{Bmatrix} = \frac{K_{II}}{G}\sqrt{\frac{r}{2\pi}}\begin{Bmatrix} -\sin\frac{\theta}{2}(\beta - 3\cos^2\frac{\theta}{2}) \\ -\cos\frac{\theta}{2}(\beta + 2 - 3\cos^2\frac{\theta}{2}) \end{Bmatrix} = K_{II}\sqrt{r}\begin{Bmatrix} f_r^b(\theta) \\ f_\theta^b(\theta) \end{Bmatrix} \tag{3}$$

Loading in Fig. 1c induces a homogeneous stress field quantified by σ_c, σ_x, and τ_c. σ_x is the normal stress component (not denoted in Fig. 1) in the direction along the fracture tip, and is typically not concerned in fracture mechanics. The displacement induced by this homogeneous stress field is

$$\begin{Bmatrix} u_r^c \\ u_\theta^c \end{Bmatrix} = r\begin{Bmatrix} f_r^c(\theta, \sigma_c, \sigma_x, \tau_c) \\ f_\theta^c(\theta, \sigma_c, \sigma_x, \tau_c) \end{Bmatrix} \tag{4}$$

or

$$\begin{Bmatrix} u_r^c \\ u_\theta^c \end{Bmatrix} = r \begin{Bmatrix} f_r^c(\theta) \\ f_\theta^c(\theta) \end{Bmatrix} \tag{5}$$

for any known stress state $(\sigma_c, \sigma_x, \tau_c)$. The explicit expression of functions f_r^c and f_θ^c can be derived based on Hooke's law, but it requires knowledge of the stress state and is not pursued here. Note that the f^c terms also encompass the effects of small rigid-body rotation of the system, but this is not explicitly discussed in the following development. The most important implication of Eq. (5) for the scope of this paper is that along any "ray" direction originating from the fracture tip, the displacement of any point relative to that of the tip is linearly proportional to its distance to the crack tip under the homogeneous stress condition.

Combining Eqs. (2), (3), and (5), we can write the overall displacement field for the arbitrary loading condition in Fig. 1 as

$$\begin{Bmatrix} u_r \\ u_\theta \end{Bmatrix} = \sqrt{r} \begin{Bmatrix} f_r^a K_I + f_r^b K_{II} \\ f_\theta^a K_I + f_\theta^b K_{II} \end{Bmatrix} + r \begin{Bmatrix} f_r^c \\ f_\theta^c \end{Bmatrix} \tag{6}$$

with K_I and K_{II} being the unknowns while u_r and u_θ can be obtained from FEM solutions.

In order to apply any displacement-based stress intensity calculation method, the medium containing the fracture needs to be modeled using a finite element mesh. Quarter-point elements, with the inverse square root singularity embedded by shifting the mid-edge nodes on the ray edges to the quarter-points, are usually employed as the first layer of elements around the tip as shown in Fig. 2. Displacements along the crack face ($\theta = \pi$) at nodes A and B are obtained by solving the finite element model. Noticing that $f_r^a(\pi) = 0$ and $f_\theta^b(\pi) = 0$, we have

$$u_r^A = \frac{1}{2}\sqrt{l_E} f_r^b(\pi) K_{II} + \frac{1}{4} l_E f_r^c(\pi) \tag{7}$$

$$u_r^B = \sqrt{l_E} f_r^b(\pi) K_{II} + l_E f_r^c(\pi) \tag{8}$$

$$u_\theta^A = \frac{1}{2}\sqrt{l_E} f_\theta^a(\pi) K_I + \frac{1}{4} l_E f_\theta^c(\pi) \tag{9}$$

$$u_\theta^B = \sqrt{l_E} f_\theta^a(\pi) K_I + l_E f_\theta^c(\pi) \tag{10}$$

where l_E is the length of the element edge ($l_E = |TB| = 4|TA|$ in this particular case). By applying basic linear equation manipulation/solving techniques, we can eliminate the terms involving f_r^c or f_θ^c and obtain

$$K_I = \frac{4u_\theta^A - u_\theta^B}{\sqrt{l_E} f_\theta^a(\pi)} \quad \text{and} \quad K_{II} = \frac{4u_r^A - u_r^B}{\sqrt{l_E} f_r^b(\pi)} \tag{11a}$$

which is the core formulation for the displacement correlation method. The symmetry of the system can be exploited to improve the accuracy of the results with

$$K_I = \frac{4(u_\theta^A - u_\theta^{A'}) - (u_\theta^B - u_\theta^{B'})}{2\sqrt{l_E} f_\theta^a(\pi)} \quad \text{and} \quad K_{II} = \frac{4(u_r^A - u_r^{A'}) - (u_r^B - u_r^{B'})}{2\sqrt{l_E} f_r^b(\pi)} \tag{11b}$$

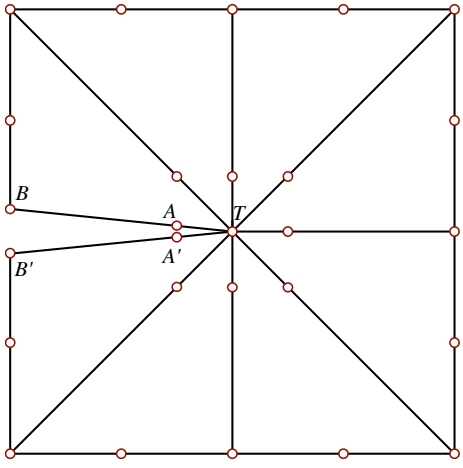


Fig. 2. Quarter-point element configurations near a crack tip.

The formulation for the so-called quarter-point displacement method

$$K_I = \frac{u_\theta^A - u_\theta^{A'}}{\sqrt{l_E} f_\theta^a(\pi)} \quad \text{and} \quad K_{II} = \frac{u_r^A - u_r^{A'}}{\sqrt{l_E} f_r^b(\pi)} \tag{12}$$

is valid only if the terms involving f_r^c and f_θ^c in Eq. (6) vanish, implying the loading of the system is the sum of case (a) and case (b) excluding case (c) in Fig. 1, i.e. there is no traction along the crack faces. This limitation of the quarter-point displacement method was described by Tracey [10] but has largely been neglected, as it does not apply to the typical loading conditions in mechanical engineering, where crack surface tractions are absent. Although this limitation of the quarter-point displacement method does not lead to inaccuracies in many studies comparing these two methods in the context of mechanical engineering [12,13,19,23], it is highly deleterious if the method is to be used for hydraulic fracturing modeling or similar problems. The displacement extrapolation method suffers similarly since the loading scenario shown in case (c) of Fig. 1 is not supported in the assumptions underlying that method. Based on this, we select the displacement correlation method as the basis for further development.

The original development of the displacement correlation method and the quarter-point displacement method derive the same equations as Eqs. (11) and (12), respectively, through a different procedure. The purpose of the above development is to provide the necessary basis for the development of the new generalized method in the next section.

3. Formulation of the generalized method

From the procedure in Section 2, we see that the core of the displacement correlation method is to solve equations of the following form

$$u_i = f_i^a \sqrt{r_i} K_I + f_i^b \sqrt{r_i} K_{II} + r_i f_i^c \tag{13}$$

where u_i, f_i^a, and f_i^b are known from FEM solutions of the specific fracture-load configuration and near-tip region closed-form solutions; K_I and K_{II} are the two unknowns to solve; f_i^c can be removed by the following procedure. Because f_i^c is a function of the angular coordinate θ but not the radial coordinate r, we can use known displacements (either u_r or u_θ) and other information (r_i, f_i^a, and f_i^b) at two points with the same angular coordinate θ to eliminate the f_i^c term. The symmetry and/or skew-symmetry of f_i^a and f_i^b can also be used to directly eliminate K_I or K_{II} when solving for the other. The choice of the four displacement components in obtaining Eqs. (7)–(10), namely $u_r^A = u_r(l_E/4, \pi)$, $u_r^B = u_r(l_E, \pi)$, $u_\theta^A = u_\theta(l_E/4, \pi)$, and $u_\theta^B = u_\theta(l_E/4, \pi)$ allows this approach. $r_i = l_E/4$ and $r_i = l_E$ are used for convenience to exploit nodal displacements in the quarter-point elements. However, displacements at other points (not necessarily nodes) can be used instead to solve Eq. (13).

Through this generalization of the original displacement correlation method, the special quarter-point element and near-tip region mesh refinement can be eliminated, and we can substitute the displacements at appropriate reference points and other necessary information into Eq. (13) to solve for SIF's. In the selection of the reference points, we first consider points with $\theta = \pm\pi$, consistent with the original displacement correlation method, where the features of $f_r^a(\pi) = 0$ and $f_\theta^b(\pi) = 0$ simplifies solution. If quadratic elements (i.e. shape functions are second-degree polynomials) are used, we can use $r = l_E/2$ and $r = l_E$, which are both within the first layer of elements about the crack tip. Appealing to symmetry, we have

$$u_r(l_E/2, \pi) - u_r(l_E/2, -\pi) = \sqrt{2l_E} f_r^b(\pi) K_{II} + \frac{l_E}{2}\left[f_r^c(\pi) - f_r^c(-\pi)\right] \tag{14a}$$

$$u_r(l_E, \pi) - u_r(l_E, -\pi) = 2\sqrt{l_E} f_r^b(\pi) K_{II} + l_E\left[f_r^c(\pi) - f_r^c(-\pi)\right] \tag{14b}$$

$$u_\theta(l_E/2, \pi) - u_\theta(l_E/2, -\pi) = \sqrt{2l_E} f_\theta^a(\pi) K_I + \frac{l_E}{2}\left[f_\theta^c(\pi) - f_\theta^c(-\pi)\right] \tag{14c}$$

$$u_\theta(l_E, \pi) - u_\theta(l_E, -\pi) = 2\sqrt{l_E} f_\theta^a(\pi) K_I + l_E\left[f_\theta^c(\pi) - f_\theta^c(-\pi)\right] \tag{14d}$$

Solving the above equations yield the formulation for the generalized displacement correlation (GDC) method as

$$K_I = \frac{2u_\theta(l_E/2, \pi) - 2u_\theta(l_E/2, -\pi) - u_\theta(l_E, \pi) + u_\theta(l_E, -\pi)}{(2\sqrt{2} - 2)\sqrt{l_E} f_\theta^a(\pi)} \tag{15}$$

$$K_{II} = \frac{2u_r(l_E/2, \pi) - 2u_r(l_E/2, -\pi) - u_r(l_E, \pi) + u_r(l_E, -\pi)}{(2\sqrt{2} - 2)\sqrt{l_E} f_r^b(\pi)} \tag{16}$$

where the constants $f_\theta^a(\pi) = f_r^b(\pi) = -\beta/\sqrt{2\pi}G$ follow from Eqs. (1)–(3). This set of equations does not require quarter-point elements around the crack tip, but does require quadratic elements (6-node triangle or 8-node quadrilateral in 2D). Since the objective of this paper is to generalize the displacement correlation method, we further consider finite element models where linear elements (3-node triangle or 4-node quadrilateral) are used. Under this condition, Eqs. (15) and (16) result in zero SIF's owing to the linear shape functions. Using displacements across two layers of elements around the tip (i.e. at $r = l_E$ and $r = 2l_E$) and replacing $l_E/2$ in the above equations with l_E and l_E with $2l_E$ solve this problem, but renders the method impractical for modeling fractures with arbitrary paths. Fig. 3 shows two problematic scenarios commonly addressed through FEM modeling of fractures: (a) sawtooth-shaped fractures typical in perturbed meshes where minor perturbation

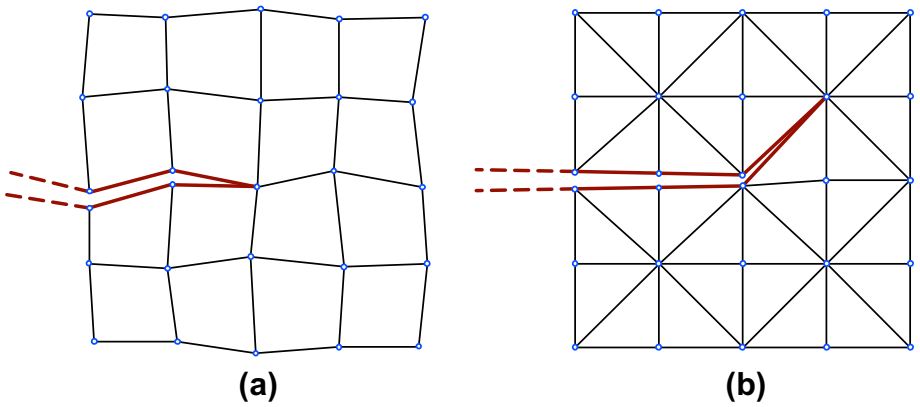


Fig. 3. Two common scenarios where the locations of points $(2l_E, \pi)$ and $(2l_E, -\pi)$ are ambiguous.

to node locations in the undeformed mesh is adopted to introduce randomness into fracture paths, and (b) a fracture having changed the direction of propagation. In both scenarios, the locations of points $(2l_E, \pi)$ and $(2l_E, -\pi)$ are ambiguous, making the method above inapplicable. To address this, we use displacements of points with $\theta = -\pi/2$, 0, and $\pi/2$ and $r = l_E$ and $r = 2l_E$, and also exploit the symmetry of f_r^a and f_θ^b and skew-symmetry of f_θ^a and f_r^b to obtain

$$u_r(l_E, \pi/2) + u_r(l_E, -\pi/2) = 2\sqrt{l_E} f_r^a(\pi/2) K_I + l_E[f_r^c(\pi/2) + f_r^c(-\pi/2)] \tag{17a}$$

$$u_r(2l_E, \pi/2) + u_r(2l_E, -\pi/2) = 2\sqrt{2l_E} f_r^a(\pi/2) K_I + 2l_E[f_r^c(\pi/2) + f_r^c(-\pi/2)] \tag{17b}$$

$$u_\theta(l_E, 0) = \sqrt{l_E} f_\theta^b(0) K_{II} + l_E f_\theta^c(0) \tag{17c}$$

$$u_\theta(2l_E, 0) = \sqrt{2l_E} f_\theta^b(0) K_{II} + 2l_E f_\theta^c(0) \tag{17d}$$

which yield

$$K_I = \frac{2u_r(l_E, \pi/2) + 2u_r(l_E, -\pi/2) - u_r(2l_E, \pi/2) - u_r(2l_E, -\pi/2)}{(4 - 2\sqrt{2})\sqrt{l_E} f_r^a(\pi/2)} \tag{18}$$

$$K_{II} = \frac{2u_\theta(l_E, 0) - u_\theta(2l_E, 0)}{(2 - \sqrt{2})\sqrt{l_E} f_\theta^b(0)} \tag{19}$$

where the constants $f_r^a(\pi/2) = (2\beta - 1)/4\sqrt{\pi}G$ and $f_\theta^b(0) = (1 - \beta)/\sqrt{2\pi}G$. We term the GDC method based on Eqs. (15) and (16) "Method A", and that based on Eqs. (18) and (19) "Method B". Method B can be applied to any finite element types, and is therefore "more general" than Method A. Method A only requires displacements across one layer of elements around the tip while Method B requires two layers. Neither Method A nor Method B requires a special meshing scheme at the near-tip region, such as a mesh type or mesh resolution different from that of the remainder of the computation domain. Both methods are easy to implement in existing FEM packages. Note that the points where displacements are used in the calculation need not to be nodes of the finite element mesh.

4. Enhancement of the generalized method

Error in the calculated stress intensity factors using the GDC method can be attributed to at least two sources:

(1) The inability of the adopted finite element's shape functions to accurately represent the near-tip displacement field. The quarter-point element family was originally formulated for the very purpose of better representing the near-tip field by including a square-root term in the shape functions in the ray directions.
(2) Omission of higher-order terms in Eqs. (1) and (3). These equations are accurate at the near-tip region, where the distances to the fracture tip and other sources inducing high displacement gradient are much smaller than the length of the fracture itself. In the GDC method, displacements at distances l_E and $2l_E$ (or $l_E/2$ and l_E) are used. Therefore, error increases with the ratio of element size to the fracture length.

In order to demonstrate the accuracy of the GDC method, we use the proposed method on the simplest fracture system, i.e. a finite-length fracture in an infinite domain as shown in Fig. 4. The fracture system considered here is straight crack of length $2a$ in a 2D infinite medium. Since most FEM models can accurately represent the linear displacement field induced by the loading condition in Fig. 1c, only the loading conditions in Fig. 1a and b are combined and modeled. However, the effects of homogeneous stress fields are appropriately handled in the formulations of the GDC method, and the superposition of

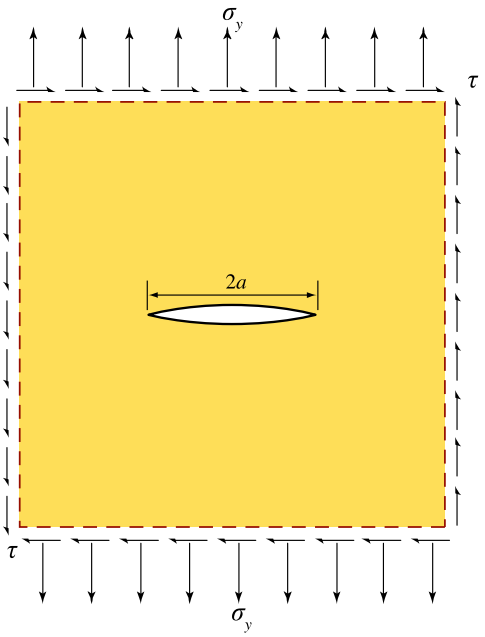


Fig. 4. A finite-length crack in an infinite medium.

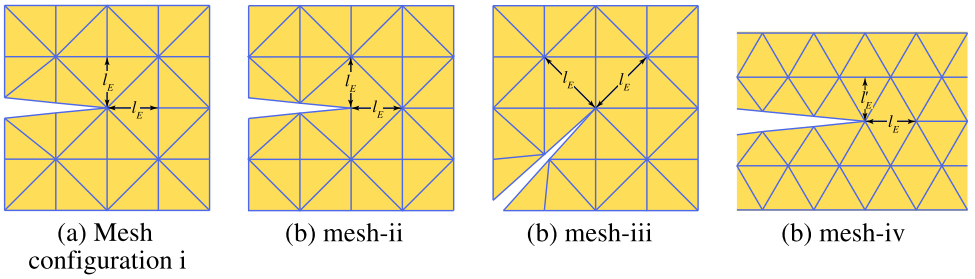


Fig. 5. Four mesh configurations considered in this study. The conventional six-node triangle element is used in all the numerical examples of the present study but the mid-edge node is not shown in this figure.

such a field would not affect the calculated SIF's. The near-tip mesh configuration can have a considerable effect on the accuracy of the original displacement-based methods (e.g. [23]); in all the numerical examples in the current and next section, the mesh configuration shown in Fig. 5a is used, and fracture tips are located at nodes shared by eight triangular elements. The other mesh configurations shown in Fig. 5 will be investigated in Section 6. In linearly elastic problems, the shear modulus of the medium, G does not affect the calculated stress intensity factors and thus can be arbitrarily selected. The model is assumed to be a plain-stress problem with a Poisson's ratio of 0.2. The effects of the Poisson's ratio will also be discussed in Section 6. The finite element mesh is sufficiently large (with each dimension longer than $100a$) such that the effects of the finite boundaries are minimal and the domain can be considered infinite. We use quadratic (6-node) triangle elements with full-integration (three Gaussian points) for both Methods A and B in this study, although Method B is not restricted to quadratic elements.

The theoretical solutions for the stress intensity factors in this crack configuration are $K_I = \sqrt{\pi a}\sigma_y$ and $K_{II} = \sqrt{\pi a}\tau$. Numerical solutions of the SIF's, denoted by K'_I and K'_{II} are obtained by solving finite element models with various levels of mesh resolutions (quantified by a/l_E, the ratio of the half crack length to element length) and substituting the obtained displacement values into Eqs. (15) and (16) or (18) and (19). We then seek an enhancement measure in the form of a "correction multiplier" to be added to Eqs. (15), (16), (18), and (19). We will test the performance of the corrected/enhanced formulation on a number of more complex crack systems in next section for Methods A and B. The values of $C_I = K_I/K'_I$ and $C_{II} = K_{II}/K'_{II}$, which are the multipliers that need to be applied to Eqs. (15) and (16) or (18) and (19), respectively to correct the numerical solutions are shown in Fig. 6 as functions of a/l_E. The correction factors are significantly larger than unity, since the 6-node triangular finite element cannot accurately represent the near-tip displacement field. C_I and C_{II} both converge to constant values as the element size becomes smaller relative to the crack length. We can fit the discrete data points with the following empirical relationship

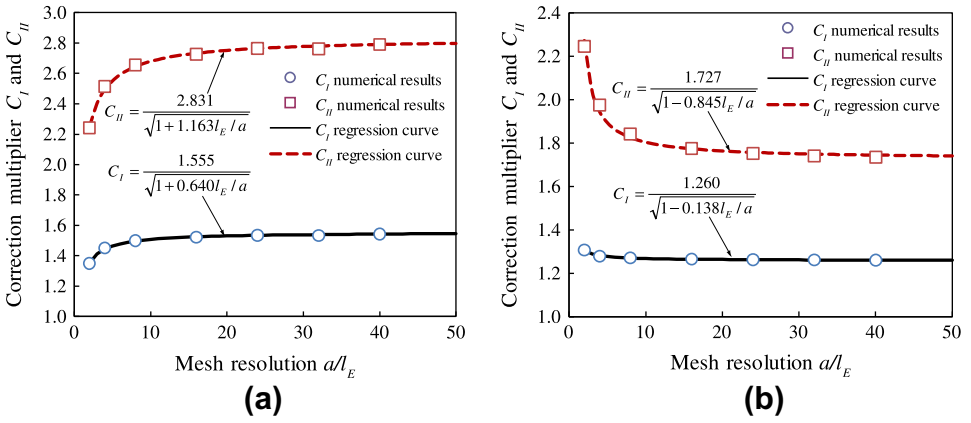


Fig. 6. The effects of the mesh resolution on the correction multipliers. (a) Results for Method A; (b) results for Method B.

$$C = \frac{\alpha_1}{\sqrt{1-\alpha_2 l_E/a}} \tag{20}$$

which has a similar format as the correction term used in [24]. The regression results are

$$C_I^A = \frac{1.555}{\sqrt{1+0.640 l_E/a}} \tag{21a}$$

$$C_{II}^A = \frac{2.831}{\sqrt{1+1.163 l_E/a}} \tag{21b}$$

$$C_I^B = \frac{1.260}{\sqrt{1-0.138 l_E/a}} \tag{21c}$$

$$C_{II}^B = \frac{1.727}{\sqrt{1-0.845 l_E/a}} \tag{21d}$$

where the superscripts A and B of C_I and C_{II} indicate whether the correction multipliers are for Method A or Method B. The coefficients of determination (R^2) for all regressions are greater than 0.99.

The correction multipliers calculated using Eq. (21) converge but not to unity. This appears counterintuitive because even though the shape functions (quadratic for the above calculations and linear if linear elements were used) of a single element does not accommodate the square root terms in Eq. (6), refining the mesh (with smaller l_E) should result in piecewise quadratic shape functions for the mesh as a whole better representing the displacement field. However, regardless of the refinement level, only displacements within the first one (Method A) or two (Method B) layers of elements around the fracture tip are used. As the mesh is refined, the reference points where displacement information is used in the calculation are closer to the fracture tip. For infinitesimal elements, this mechanism can eliminate the error induced by the second source of error, but not the first. A similar phenomenon exist for the original displacement-based methods: Numerous studies have observed that errors of these methods do not converge to zero as the near-tip mesh is refined [12,13,18,19,23] and an explanation was offered by Harrop [25].

5. Accuracy of the generalized method for different fracture configurations

The values as well as the regression formula of the correction multipliers in Section 4 are obtained for a specific fracture-load configuration. Considering that the main purpose of this correction term is to correct errors caused by the finite elements' inability to accurately represent the near-tip displacement field described by Eqs. (1)–(3), we hypothesize that the same multipliers can be applied to all other crack-load configurations and obtain reasonable SIF results. In this section, we apply the correction multipliers obtained from the special case in Section 4 to a spectrum of fracture configurations to test this hypothesis. Special attention is paid to coarse meshes and effects of interference between neighboring fractures and between fractures and free surfaces. Achieving acceptable accuracy under these conditions is crucial for managing the computational cost of the simulation of dynamic fracture propagation in complex fracture systems. Four test cases for which closed-form solutions of SIF's exist are carefully selected: The first case embodies the interference between fracture tip and free surface boundary; the second deals with heterogeneous stress field; the last two cases represent interactions between neighboring fractures. Both mode-I and mode-II SIF's are considered whenever applicable. Both Method A and Method B are

evaluated for the first case in Section 5.1. Since the mathematical and mechanical principles behind these two methods are similar, only the more general Method B is considered for the other three fracture-load configurations.

5.1. Center-cracked infinite strip with a finite width

Consider a center-cracked strip with an infinite length but finite width $2b$. The crack is $2a$ long and perpendicular to the longitudinal direction of the strip as shown in Fig. 7a. The strip is subjected to a tensile stress σ in the longitudinal direction and a uniformly distributed shear stress τ along the fracture faces, inducing mode-I and mode-II stress concentration, respectively. The stress intensity factors are

$$K_I = \sigma\sqrt{\pi a}F_I(a/b) \quad \text{and} \quad K_{II} = \tau\sqrt{\pi a}F_{II}(a/b) \tag{22}$$

where F_I and F_{II} are the fracture-configuration correction factors that can be estimated using the modified Koiter's formula [1]:

$$F_I(a/b) = F_{II}(a/b) = [1 - 0.025(a/b)^2 + 0.06(a/b)]\left(\cos\frac{\pi a}{2b}\right)^{-1/2} \tag{23}$$

with a relative error of less than 0.1% for any a/b value. In this and other examples, if F_I and F_{II} are close to unity, it means this fracture-load configuration is similar to the reference configuration of a single fracture in an infinite plane.

To apply the GDC method, the strip is discretized into a finite element mesh of a length that is more than 12 times longer than its width, which is found to sufficiently approximate the infinite length according to a sensitivity analysis. Different levels of mesh refinement with b/l_E ranging from 4 to 64 as well as various crack length-to-strip width ratios, i.e., $a/b = 0.125$, 0.25, 0.50, 0.75, and 0.875 are adopted to investigate the effects of these two factors. Due to the symmetry of the crack and mesh configuration, the tensile stress σ does not contribute to the calculated K_{II} and τ does not contribute to K_I. In all the numerical examples in Section 4, a Poisson's ratio of 0.2 and the crack tip mesh configuration shown in Fig. 5a (eight triangle elements connected to the tip) are used. The effects of the Poisson's ratio and crack tip mesh configuration will be studied in Section 6. To allow precise comparison, the calculation results of the GDC method (both Method A and Method B) with the correction multipliers computed using Eq. (21) applied, as well as the theoretical solution based on Eq. (23) are shown in Tables 1A–2B. Note that the values of F_I and F_{II}, instead of the stress intensity factors K_I and K_{II} are shown. F_I and F_{II} can be considered normalized values of the SIF's. Due to the relationships described in Eq. (22), the relatively errors for K_I and K_{II} are the same as those for F_I and F_{II}, respectively.

The results show that Method B for mode-I fracturing and Method A for both mode-I and -II are fairly accurate for all the scenarios considered, including those with very coarse meshes. The relative errors are generally smaller than 2% with few exceptions. The accuracy of Method-B for mode-II fracturing seems to be dependent on the fracture geometry and mesh resolution. For $b/l_E = 4$ with $a/b = 0.5$, $b/l_E = 8$ with $a/b = 0.75$, and $b/l_E = 16$ with $a/b = 0.875$, erroneous results are obtained.

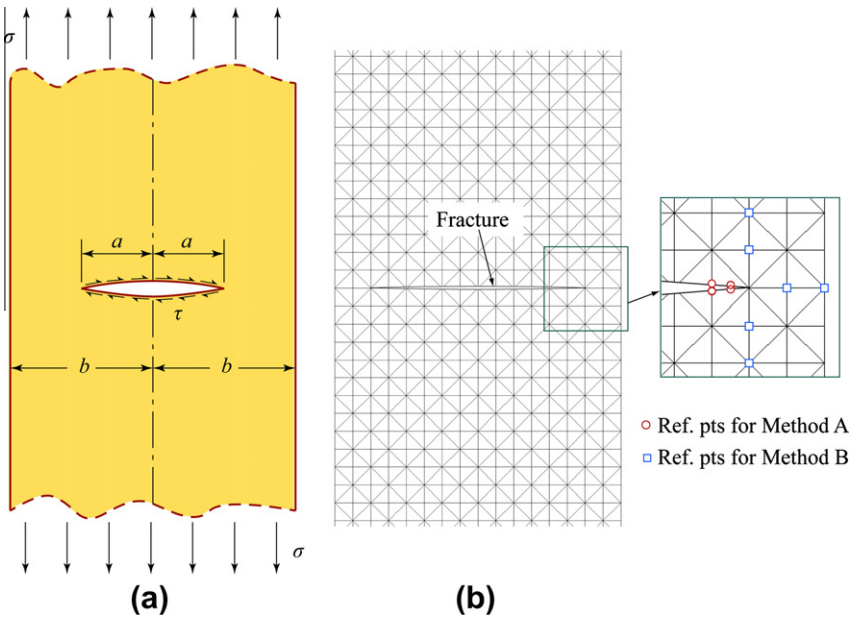


Fig. 7. Center-cracked infinite strip with a finite width. (a) The crack configuration; (b) the mesh for the case where $b = 8l_E$ and $a/b = 0.75$ (with opening of the fracture exaggerated). The reference points used by Method A and Method B are indicated in the figure.

Table 1A
Calculated F_I values using the GDC method (Method A) for the center-cracked infinite strip.

a/b	F_I, numerical result					Relative error (%)					$F_I(a/b)$ Eq. (23)
	$b/l_E = 4$	8	16	32	64	$b/l_E = 4$	8	16	32	64	
0.125	N/A[a]	N/A[a]	1.011	1.004	1.007	N/A[a]	N/A[a]	0.1	−0.6	−0.2	1.009
0.25	N/A[a]	1.038	1.032	1.036	1.040	N/A[a]	−0.1	−0.7	−0.3	0.1	1.039
0.50	1.168	1.171	1.179	1.186	1.189	−1.5	−1.3	−0.6	0.0	0.3	1.186
0.75	N/A[a]	1.595	1.612	1.622	1.628	N/A[a]	−1.8	−0.8	−0.1	0.2	1.624
0.875	N/A[a]	N/A[a]	2.271	2.288	2.300	N/A[a]	N/A[a]	−1.3	−0.5	0.0	2.300

[a] N/A, numerical results unavailable due to the incompatibility between the a/b value and the mesh configuration.

Table 1B
Calculated F_I values using the GDC method (Method B) for the center-cracked infinite strip.

a/b	F_I, numerical result					Relative error (%)					$F_I(a/b)$ Eq. (23)
	$b/l_E = 4$	8	16	32	64	$b/l_E = 4$	8	16	32	64	
0.125	N/A	N/A	1.008	1.011	1.009	N/A	N/A	−0.1	0.2	0.0	1.009
0.25	N/A	1.036	1.040	1.038	1.037	N/A	−0.3	0.1	−0.1	−0.1	1.039
0.50	1.196	1.186	1.182	1.183	1.184	0.8	0.0	−0.4	−0.3	−0.2	1.186
0.75	N/A	1.640	1.618	1.617	1.619	N/A	1.0	−0.4	−0.5	−0.3	1.624
0.875	N/A	N/A	2.325	2.295	2.291	N/A	N/A	1.1	−0.2	−0.4	2.300

Table 2A
Calculated F_{II} values using the GDC method (Method A) for the center-cracked infinite strip.

a/b	F_{II}, numerical result					Relative error (%)					$F_{II}(a/b)$ Eq. (23)
	$b/l_E = 4$	8	16	32	64	$b/l_E = 4$	8	16	32	64	
0.125	N/A	N/A	1.013	1.000	1.006	N/A	N/A	0.3	−0.9	−0.3	1.009
0.25	N/A	1.040	1.030	1.038	1.045	N/A	0.1	−0.8	−0.1	0.6	1.039
0.50	1.165	1.172	1.188	1.201	1.208	−1.8	−1.2	0.2	1.2	1.8	1.186
0.75	N/A	1.579	1.621	1.645	1.658	N/A	−2.8	−0.2	1.3	2.1	1.624
0.875	N/A	N/A[a]	2.241	2.294	2.323	N/A	N/A	−2.6	−0.3	1.0	2.300

Table 2B
Calculated F_{II} values using the GDC method (Method B) for the center-cracked infinite strip.

a/b	F_{II}, numerical result					Relative error (%)					$F_{II}(a/b)$ Eq. (23)
	$b/l_E = 4$	8	16	32	64	$b/l_E = 4$	8	16	32	64	
0.125	N/A	N/A	1.021	0.994	1.001	N/A	N/A	1.1	−1.6	−0.8	1.009
0.25	N/A	1.027	1.018	1.031	1.041	N/A	−1.2	−2.0	−0.8	0.2	1.039
0.50	**0.014**[b]	0.972	1.132	1.181	1.200	**−98.8**	−18.1	−4.5	−0.4	1.2	1.186
0.75	N/A	**−0.841**[b]	1.124	1.502	1.610	N/A	**−152**	−30.8	−7.6	−0.9	1.624
0.875	N/A	N/A[a]	**−1.710**[b]	1.432	2.070	N/A	N/A	**−174**	−37.8	−10.0	2.300

[b] Degenerate results; see discussion below. The Bold typeface used in other tables highlights degenerate results owing to similar reasons.

In these three situations, the fracture tip is two elements away (i.e. $(b-a)/l_E = 2$) from the lateral boundary. One of the displacement components used in Eq. (19), $u_\theta(2l_E, 0)$ happens to be at the lateral boundary. The mechanical response at this point is substantially affected by the free-surface boundary condition and violate an assumption of the GDC method. This is not an issue for Method A or the calculation of K_I using Method B because none of the displacement components used in Eqs. 15, 16, and (18) is at the boundary. At the same mesh refinement level, if the distance between the crack tip and the lateral free-surface boundary is $4l_E$ instead of $2l_E$, the relative error for K_{II} (Method B) is approximately between 20% and 40%, which though suboptimal for typical mechanical engineering applications is often acceptable for geo-science or geo-engineering scenarios due to the high aleatoric uncertainty in geo-systems. Nevertheless, if the crack tip is $6l_E$ or farther away from the free surface, the error drops below 10% for K_{II} by Method B.

5.2. Three-point bending beam with a notch at mid-span

Consider a beam specimen with a span-to-height ratio of $s/b = 4$ with a notch of length a cut at the mid-span as shown in Fig. 8. The beam is subjected to a mid-span force P. Due to the symmetry of the configuration, the mode-II stress intensity factor is zero, and for mode-I

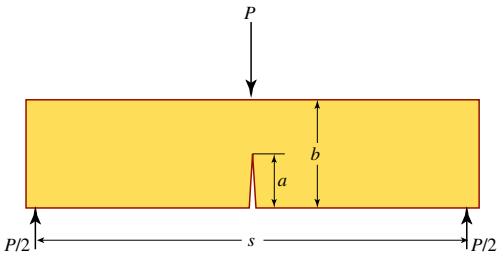


Fig. 8. Three-point bending beam with a mid-span notch.

Table 3
Calculated F_I values using the GDC method for the three-point bend beam (Method B only).

a/b	F_I, numerical result					Relative error (%)					$F_I(a/b)$ Eq. (25)
	$b/l_E = 4$	8	16	32	64	$b/l_E = 4$	8	16	32	64	
0.125	N/A	N/A	0.944	0.965	0.972	N/A	N/A	−5.1	−3.0	−2.3	0.995
0.25	N/A	1.013	1.005	1.003	1.001	N/A	0.5	−0.2	−0.4	−0.6	1.007
0.50	1.581	1.468	1.422	1.409	1.406	11.7	3.7	0.4	−0.5	−0.7	1.416
0.75	N/A	3.623	3.439	3.369	3.352	N/A	8.2	2.7	0.6	0.1	3.349
0.875	N/A	N/A	9.469	9.075	8.929	N/A	N/A	7.1	2.6	1.0	8.843

$$K_I = \frac{3Ps}{2b^2}\sqrt{\pi a}F_I(a/b) \tag{24}$$

where $F_I(a/b)$ is the fracture-configuration correction factor, with similar meaning to its counterpart in Eq. (22) but different values. Its value can be calculated using the following dimensionless regression equation proposed by Srawley [26] with a relative error smaller than 0.5%

$$F(a/b) = \frac{1.99 - a/b(1-a/b)[2.15 - 3.93a/b + 2.7(a/b)^2]}{(1+2a/b)(1-a/b)^{3/2}\sqrt{\pi}} \tag{25}$$

To test the accuracy of the GDC method on this configuration, we perform FEM analysis with different levels of mesh refinement and different notch lengths. The results of Method-B are summarized in Table 3 in a manner similar to that of Tables 1 and 2. The results are generally accurate. In the worst case scenario, where the height direction of the beam is discretized into four element, the relative error is 11.7%, which remains acceptable for many engineering applications. As the mesh is refined, the numerical results for each geometrical configuration generally converge to the closed-form solution with some minor fluctuation (a few tenths of a percent), which is within the 0.5% error inherent in the closed-form solution. The accuracy is compromised when the notch is short or long compared with the beam height (e.g. $a/b = 0.125$ or 0.875). In both cases, the points where the displacements are used in the GDC method have similar distances to the notch tip and to the lower or upper free surface of the beam and are not within the near-tip region.

5.3. Two finite-length fractures along a single line

In Sections 5.3 and 5.4, we investigate the accuracy of the GDC method for scenarios with multiple fractures interacting with each other. We first consider the configuration shown in Fig. 9, where two finite-length fractures along a single line existing in an infinite plane. This configuration tends to strengthen the stress intensity at the two tips A and B, compared with the configurations whether each crack exists alone in an infinite plane. For any tip under a given far-field stress condition (σ and τ), the stress intensity factors (mode-I and mode-II) are dependent on certain geometrical features of the system, and the following closed-form solutions are available [1]

$$K_I^A = \sigma\sqrt{\pi b}F_I^A(a/b, c/b) \tag{26a}$$

$$K_{II}^A = \tau\sqrt{\pi b}F_{II}^A(a/b, c/b) \tag{26b}$$

$$K_I^B = \sigma\sqrt{\pi a}F_I^B(a/b, c/b) \tag{26c}$$

$$\text{and} \quad K_{II}^B = \tau\sqrt{\pi a}F_{II}^B(a/b, c/b) \tag{26d}$$

where

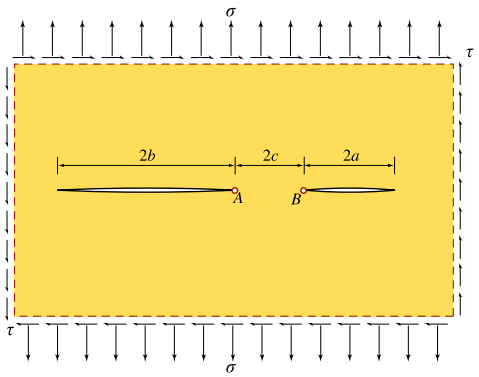


Fig. 9. Two finite-length fractures along a single line in an infinite plane.

Table 4
Calculated stress intensity for the two-fracture case at crack tip *A* (Method B only).

b/c	F_I, numerical result			F_I, relative error (%)			F_{II}, numerical result			F_{II}, relative error (%)			F_I, F_{II} anly. solu.
	$b/l_E = 4$	8	16	$b/l_E = 4$	8	16	$b/l_E = 4$	8	16	$b/l_E = 4$	8	16	
2	1.027	1.036	1.041	−1.5	−0.6	−0.2	0.975	1.015	1.035	−6.5	−2.7	−0.7	1.043
4	1.044	1.071	1.090	−5.1	−2.7	−0.9	**0.318**	1.004	1.068	**−71.1**	−8.7	−2.9	1.100
8	*1.137*[c]	1.113	1.163	*−5.6*[c]	−7.7	−3.5	N/A	**0.246**	1.076	N/A	**−79.6**	−10.7	1.206
16	N/A	*1.249*[c]	1.248	N/A	*−9.3*[c]	−9.3	N/A	N/A	**0.185**	N/A	N/A	**−86.5**	1.377
32	N/A	N/A	*1.445*[c]	N/A	N/A	*−11.4*[c]	N/A	N/A	N/A	N/A	N/A	N/A	1.632

$$F_I^A = F_{II}^A = \frac{1}{\sqrt{1-\alpha_A}}\left[1 - \frac{1}{\alpha_B}\left\{1 - \frac{E(k)}{K(k)}\right\}\right] \tag{27a}$$

$$F_I^B = F_{II}^B = \frac{1}{\sqrt{1-\alpha_B}}\left[1 - \frac{1}{\alpha_A}\left\{1 - \frac{E(k)}{K(k)}\right\}\right] \tag{27b}$$

with $\alpha_A = a/(a+c)$, $\alpha_B = b/(b+c)$, and $k = \sqrt{\alpha_A \alpha_B}$ and

$$K(k) = \int_0^{\pi/2} (1 - k^2 \sin^2 \varphi)^{-1/2} d\varphi \tag{28a}$$

$$E(k) = \int_0^{\pi/2} (1 - k^2 \sin^2 \varphi)^{1/2} d\varphi \tag{28b}$$

In the numerical solutions, we fix the length ratio of the two fractures to be $a/b = 0.5$ and investigate the effects of the mesh refinement levels ($b/l_E = 4$, 8, and 16) and the distance between the two fracture tips ($c/b = 1/2$, 1/4, 1/8, and 1/16 whenever applicable). The finite element model is more than $50b$ long in each dimension to minimize the boundary effects. The numerical results for the two crack tips *A* and *B* are summarized in Tables 4 and 5, respectively.

The trends observed in this series of results are similar to those from Sections 5.1 and 5.2. Method B of the GDC method is more accurate for mode-I stress intensity than for mode-II. Even under pathological conditions, i.e. mesh coarseness limit reached and strong numerical coupling between the two tips, the error is of the order of 10%. The accuracy for mode-II is non-ideal but still acceptable for many applications. The only exceptions are when the two tips are only two elements away from each other. In this situation, $u_\theta(2l_E, 0)$ used in Eq. (19) for a tip is the displacement of the other tip, resulting in strong numerical coupling between the two fractures. In these situations, Method A is more appropriate since it uses displacements "behind" fracture tips, where less numerical coupling between the two fractures is expected.

5.4. An infinite array of parallel fractures in an infinite plane

Consider the fracture configuration shown in Fig. 10 where an infinite array of parallel finite-length cracks are periodically arranged on an infinite plane subjected to far-field stress. The interaction between fractures tends to reduce mode-I stress intensity but enhance mode-II stress intensity. The stress intensity factors are $K_I = \sigma\sqrt{\pi a}F_I(a/h)$ and $K_{II} = \tau\sqrt{\pi a}F_{II}(a/h)$ where F_I and F_{II} are the crack configuration correction factors as functions of the crack length and the interval between neighboring cracks. The analytical solutions for F_I and F_{II} are unavailable but well-accepted numerical solutions are presented in [1] and are plotted as continuous curves in Fig. 11. In the FEM solution of this study, we investigate the effects of mesh refinement level ($a/l_E = 16$, 8, 4, and 2) and distance between adjacent fractures (a/h). Due to the periodicity of the

Table 5
Calculated stress intensity for the two-fracture case at crack tip *B* (Method B only).

b/c	F_I, numerical result			F_I, relative error (%)			F_{II}, numerical result			F_{II}, relative error (%)			F_I, F_{II} anly. solu.
	$b/l_E = 4$	8	16	$b/l_E = 4$	8	16	$b/l_E = 4$	8	16	$b/l_E = 4$	8	16	
2	1.078	1.113	1.122	−4.2	−1.1	−0.3	0.978	1.058	1.098	−13.1	−6.0	−2.5	1.126
4	1.117	1.197	1.238	−11.2	−4.8	−1.5	**−0.526**	1.038	1.177	**−142**	−17.4	−6.3	1.257
8	*1.304*[c]	1.287	1.387	*−10.9* [c]	−12.1	−5.2	N/A	**−0.370**	1.204	N/A	**−125**	−17.7	1.464
16	N/A	*1.533*[c]	1.541	N/A	*−12.9* [c]	−12.5	N/A	N/A	**−0.270**	N/A	N/A	**−115**	1.761
32	N/A	N/A	*1.878* [c]	N/A	N/A	*−13.4* [c]	N/A	N/A	N/A	N/A	N/A	N/A	2.169

[c] Limit of the mesh coarseness reached where only one element exist between tip *A* and tip *B*. K_{II} cannot be calculated at this level of mesh refinement using Method B.

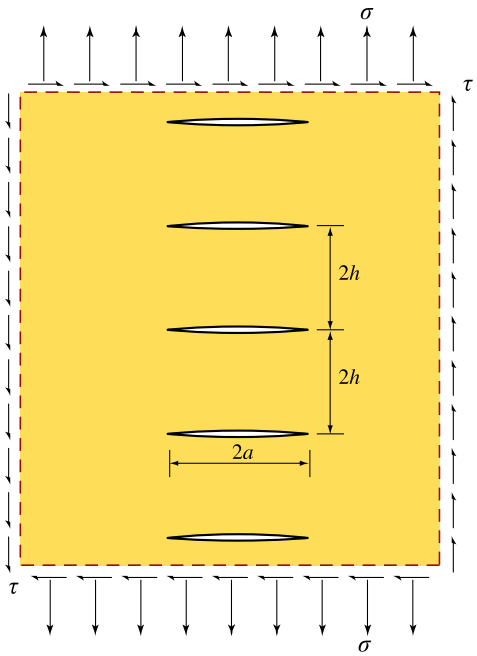


Fig. 10. Parallel finite-length fractures in an infinite plane.

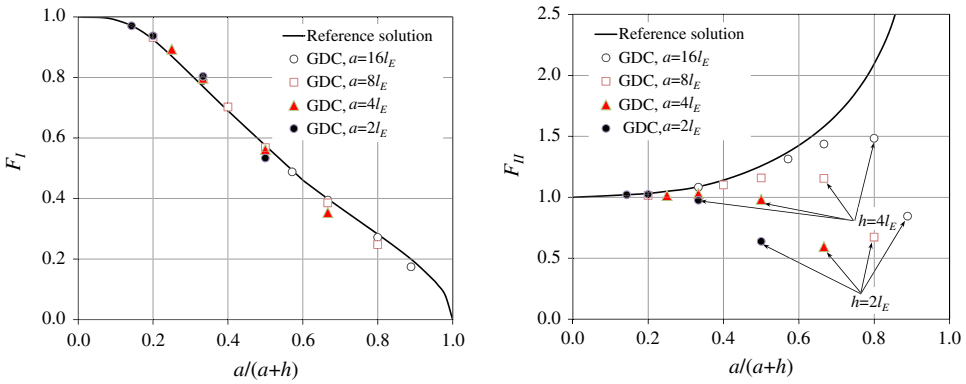


Fig. 11. Comparison of the GDC method results and well-accepted reference numerical solutions [1]. The latter are shown as continuous curves and they have an estimated error of less than 1%. $a/(a+h)$ is used as the horizontal axis to be consistent with the notation in [1]. Note that $a/(a+h) = 1/(1+h/a)$.

configuration, only one crack and the surrounding medium need to be included in the mesh with appropriate periodic boundary conditions applied. The width of the mesh is more than 50 times the crack length to minimize the effects of the far-field lateral boundaries. As shown in Fig. 11, the results of the GDC methods (Method B only) are fairly accurate for mode-I with relative errors below 10%. The results for mode-II are less accurate and the most significant factor affecting

the accuracy is h/l_E. When $h/l_E = 4$ (i.e. eight elements between adjacent cracks), the relative error can be as high as 30% for large a/h values, but the ascending trend of the $F_{II} - a/(a+h)$ curve can still be reproduced. When $h/l_E = 2$, the relative error becomes unacceptably large and fails to represent the general trend of the $F_{II} - a/(a+h)$ curve. Among all the numerical cases, the shortest distance between neighboring cracks is $4l_E$ (i.e. $h/l_E = 2$). If the neighboring cracks are only $2l_E$ apart, Method B for mode-I will fail because all the displacement components used in Eq. (18) would be zero due to the symmetry of the problem, yielding zero stress intensity. This condition dictates the largest element size that can be used for mode-I.

6. The effects of mesh configurations and the Poisson's ratio

In all the numerical examples in Sections 4 and 5, the Poisson's ratio is assumed to be 0.2. As shown in Eq. (1), the Poisson's ratio is related to the value of β thereby affecting the near-tip displacement field. As mentioned in Section 3, the accuracy of the GDC method (without enhancement through the correction multipliers) depends on the ability of the finite element in representing the near-field displacement field. Therefore, it is expected that the values of C_I and C_{II} are dependent on the Poisson's ratio. We repeat the numerical examples on a single fracture in an infinite plane in Section 4 with Poisson's ratios ranging from 0 to 0.4, and the correction multipliers required for obtaining accurate SIF's for different mesh refinement levels are shown in Fig. 12. A unified regression model is established by assuming the two constants in Eq. (20) to vary linearly with respect to the Poisson's ratio, and the regression results are

$$C_I^B = \frac{1.226 + 0.206\nu}{\sqrt{1 - (0.349 - 1.125\nu)l_E/a}} \tag{29a}$$

$$C_{II}^B = \frac{1.737 - 0.048\nu}{\sqrt{1 - (0.874 - 0.179\nu)l_E/a}} \tag{29b}$$

The effects of the Poisson's ratio are more significant for mode-I than for mode-II. Even for mode-I, ignoring these effects by using the correction multipliers for $\nu = 0.2$ introduces less than 4% incremental error to the calculated SIF's for arbitrary Poisson's ratio.

The correction multipliers are also dependent on the near-tip mesh configuration. All the previous numerical examples are based on the mesh configuration shown in Fig. 5a where eight triangular elements are connected to the tip node. The other thee configurations in Fig. 5 are also common in FEM analysis. We repeat the numerical analysis in Section 4 with the additional mesh configurations to determine the correction multipliers for different configurations and the results for a Poisson's ratio of 0.2 are shown in Fig. 13. Note that mesh-i, mesh-ii, and mesh-iii use the same space discretization scheme with the only difference among them being in the location of the crack tip and the crack orientation. For a given mesh, the l_E value of mesh-iii is $\sqrt{2}$ times larger than that for mesh-i and mesh-ii. To use mesh configuration iv, l_E in Eq. (18) is replaced with $l'_E\sqrt{3}l_E/2$. This constrains the solution to only use the displacements of points within two element layers of the tip.

The trend of the variation of the correction multipliers with respect to the mesh refinement level is the same for all the mesh configurations. The curves become relatively flat when $a/l_E > 8$. In configurations i and iii, the near-tip region is discretized into eight elements in the angular direction while it is discretized into four elements for mesh-ii. Better refinement in the angular direction improves the displacement field representation, yielding correction multipliers closer to unity. In the region with a radius of $2l_E$ around the tip, more elements are involved in mesh-iii than in mesh-i (the mesh is the same for these two configurations but l_E for mesh-iii is longer), enabling a better displacement field representation. Despite these observations, the effects of the mesh configurations on the correction multipliers are moderate. If we used the correction multipliers for mesh-i on mesh configuration ii, it would induce an error of 4%.

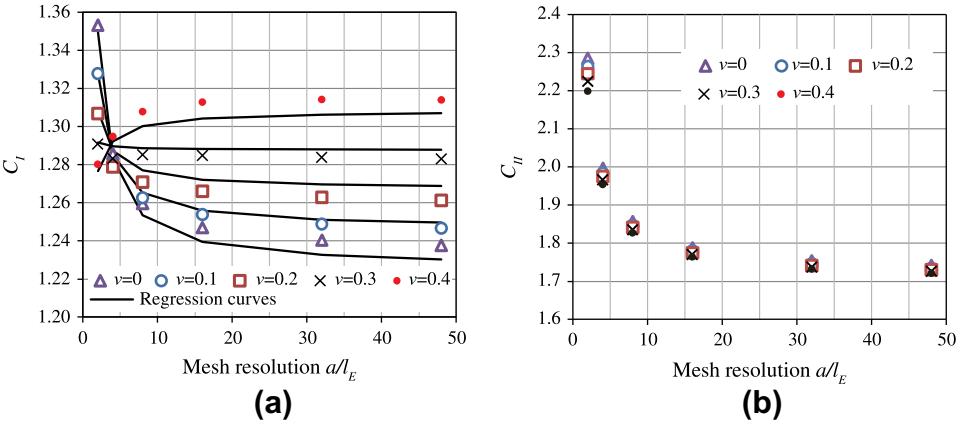


Fig. 12. The effects of the Poisson's on the correction multipliers for (a) mode-I and (b) mode-II at different mesh refinement levels. The effects on C_{II} are small and the regression curves are not plotted. Only the results for Method B are shown.

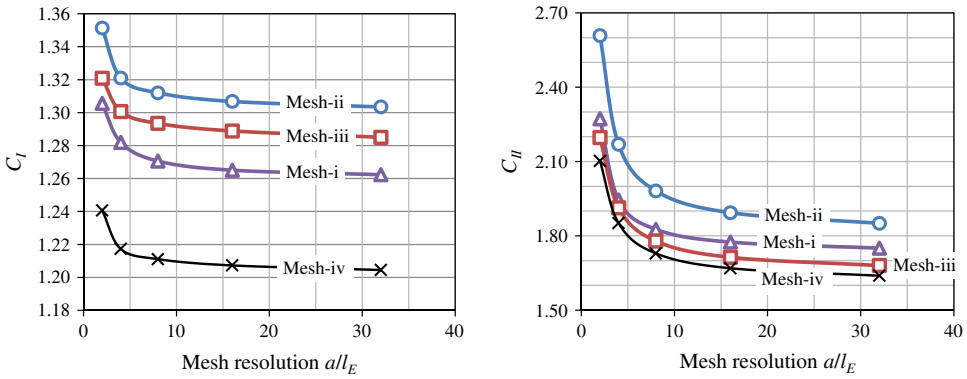


Fig. 13. The effects of near-tip mesh configurations on the correction multipliers for (a) mode-I and (b) mode-II at different mesh refinement levels.

Additionally, though all the examples in this paper are for plane-stress conditions using Method B, application of the generalized Method B to plane-strain conditions or Method A to plane-strain and plane-stress conditions is straightforward.

7. The effects of mesh perturbation

In the previous numerical examples, the finite meshes are all based on regular grids and the fractures align with the grids. In this section, we investigate the effects of mesh perturbation on the accuracy of the GDC method. Only the mesh pattern shown in Fig. 5a is tested but the qualitative observations should apply to all mesh patterns. A mesh perturbation factor δ is introduced to quantify the degree of perturbation. The x- and y-coordinates of each end-edge node is moved from its original location in the regular mesh by a distance that follows a uniform distribution between $-\delta l_E$ and δl_E. The mid-edge nodes are moved accordingly. The nodes along external boundaries and existing fractures are not perturbed in order to maintain the geometrical configurations of the system. Three levels of perturbation with $\delta = 0.1$, 0.2, and 0.3 are considered. For a given level of perturbation, different mesh patterns can be obtained by altering the seed value for the random number generator used in the meshing routine. Eight individual and independent random realizations are analyzed as a simple random sample for each perturbation level. Both Method A and Method B are evaluated in this section wherever appropriate. For the perturbed mesh, we still use the characteristic element size parameter l_E of the parent regular mesh, which is essentially the average element size in the perturbed mesh. The correction factors derived based on the regular mesh are applied.

The two fracture-loading configurations investigated in Sections 5.1 and 5.3, representing fracture-boundary and fracture-fracture interactions, respectively, are assessed. These two configurations are termed the "finite strip" case and "dual fracture" case in the following description. One relatively coarse mesh resolution is used for each case. For the finite strip scenario, $b/l_E = 16$ and $a/l_E = 12$, so the distance between the fracture tip and the lateral boundary is approximately four times the element size. One of the reference points used by Method B for mode-II is at the middle point between the fracture tip and the free surface boundary, not strictly speaking in the near-tip region. The error for this particular case is approximately 31% as shown in Table 2-B, and Method B for mode-II is inappropriate for this particular case. We present the results with mesh perturbation for this case anyway for the sake of completeness, but this limitation should be born in mind. For the dual fracture scenario, $b/l_E = 8$, $a/l_E = 4$, and $c = l_E$, so there is only two elements between the two fracture tips, which made Method B for mode-II inapplicable because one of the reference points for a fracture tip is at the tip of the other tip, as illustrated by the very poor results corresponding to this particular scenario in Table 4 (bold font). Therefore, K_{II} using Method B for the second fracture-loading configuration is not pursued here. The mesh patterns and locations of reference points are illustrated in Fig. 14.

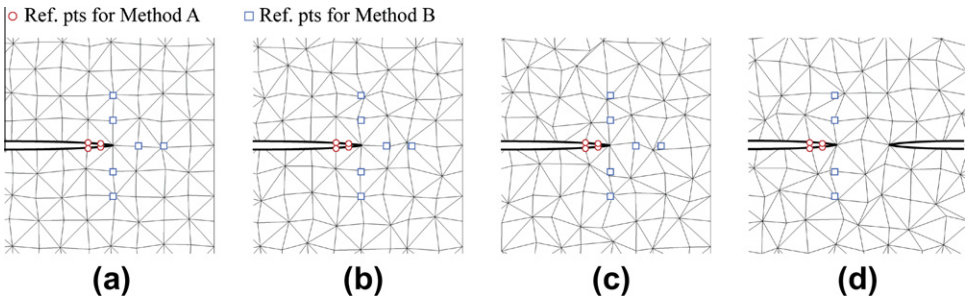


Fig. 14. Perturbed mesh and location of reference points. (a–c) are all based on the finite strip configuration in Section 5.1 and they have perturbation factor $\delta = 0.1$, 0.2, and 0.3, respectively; (d) is based on the dual fracture configuration investigated in Section 5.3 and only the mesh for $\delta = 0.3$ is shown. Note that reference points for Method B on perturbed mesh generally do not coincide with nodes.

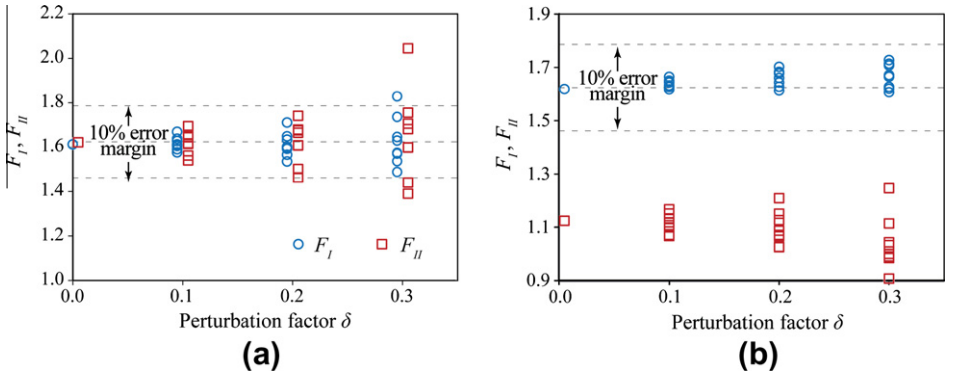


Fig. 15. Results of the GDC method for the finite strip case with perturbed meshes. (a) Results for Method A; (b) results for Method B. Each data point represents one random realization of a mesh perturbation level. The horizontal coordinates of some data points are slightly offset to enable visually separating mode-I and mode-II data points. The margins for 10% relative error are shown in this figure.

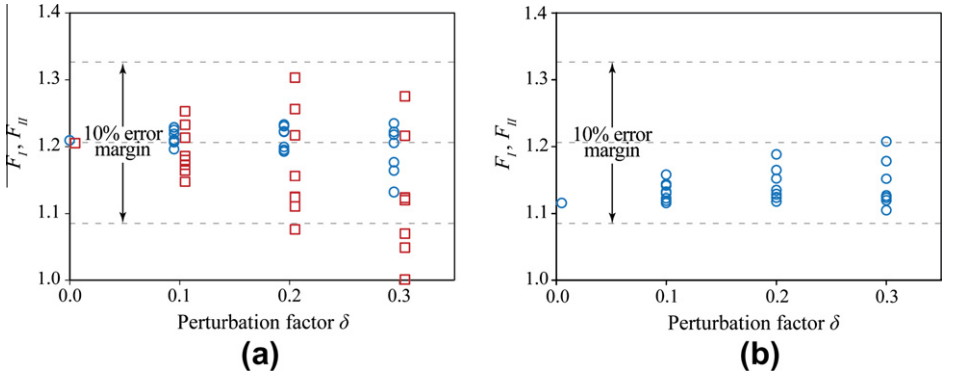


Fig. 16. Results of the GDC method for the dual fracture case with perturbed meshes. (a) Results for Method A; (b) results for method B. Results for mode-II with method B are not presented for reasons described above.

The calculation results of the GDC method for individual random realizations of different mesh perturbation levels are shown in Figs. 15 and 16, for the finite strip case and the dual fracture case, respectively. Considering the stochastic nature of mesh perturbation, we also present some statistical analysis results in Tables 6A and 6B. As expected, mesh perturbation affects the GDC calculation results in a random manner, with a greater degree of perturbation causing greater variation of SIF results. Mode-II results seem to be more sensitive to mesh perturbation that mode-I results. The mean of SIF results by Method A appears to be unaffected by mesh perturbation, whereas mesh perturbation slightly increases the mean of mode-I SIF by Method B. Note that the results of mode-II SIF by Method B for both cases tested are inherently inaccurate due to inadequate mesh resolution for these particular cases. The results for the finite strip case are shown here only to illustrate the additional errors induced by mesh perturbation. In most cases, even relatively severe mesh perturbation induces less than 10% additional error, whereas practical needs for mesh perturbation more severe than $\delta = 0.3$ are very rare. More importantly, these numerical examples demonstrate that the GDC method is reasonably robust and its general accuracy does not rely on the symmetry or regularity of meshing.

8. Concluding remarks

Compared with the original displacement-based methods for calculating stress intensity factors, the generalized displacement correlation (GDC) method proposed in this paper has two advantages: (1) It is designed to work with conventional finite element types, and (2) it uses a homogeneous mesh without local refinement around fracture tips. The former feature makes it convenient to implement the new method in existing finite element packages. The latter is important for modeling dynamic fracture propagation problems where the locations of fractures are not known *a priori*. These two features are critical to engineering applications where adopting special element types and local refinement are impractical, such as in the simulation of hydraulic fracturing in complex natural fracture systems.

We propose two suites of formulations, termed Method A and Method B, for the GDC method. The former utilizes displacement information within one layer of elements around the fracture tip, and requires quadratic or higher-order finite elements. The latter can work with any element types, but requires displacements within two layers of elements. To enhance

Table 6A
GDC results for the finite strip case with various levels of mesh perturbation.

		Theoretical	GDC results						
			$\delta = 0$	$\delta = 0.1$		$\delta = 0.2$		$\delta = 0.3$	
				Mean	St. dev.	Mean	St. dev.	Mean	St. dev.
Method A	F_I	1.624	1.612	1.616	0.029	1.611	0.054	1.626	0.111
	F_{II}	1.624	1.621	1.622	0.050	1.617	0.094	1.651	0.203
Method B	F_I	1.624	1.618	1.638	0.016	1.654	0.032	1.667	0.046
	F_{II}	1.624	1.124[d]	1.114	0.035	1.095	0.063	1.040	0.102

[d] Poor results due to inappropriate mesh resolution, as explained in Section 5.1.

Table 6B
GDC results for the dual fracture case with various levels of mesh perturbation.

		Theoretical	GDC results						
			$\delta = 0$	$\delta = 0.1$		$\delta = 0.2$		$\delta = 0.3$	
				Mean	St. dev.	Mean	St. dev.	Mean	St. dev.
Method A	F_I	1.206	1.209	1.215	0.012	1.211	0.017	1.196	0.035
	F_{II}	1.206	1.205	1.194	0.036	1.169	0.081	1.134	0.095
Method B	F_I	1.206	1.116	1.133	0.014	1.145	0.024	1.149	0.036
	F_{II}	1.206	N/A	N/A	N/A	N/A	N/A	N/A	N/A

accuracy of both methods, a correction multiplier is also proposed. Without this correction term, the accuracy of the GDC method is limited due to the inability of regular finite element types to accurately represent the near-tip displacement field. Through a series of numerical examples with a variety of crack configurations, we find that that the new GDC method is acceptably accurate for calculating mode-I stress intensity factors. Even in the limit of mesh coarseness when there is only one element between the two tips of the adjacent fractures, the error is of the order of 10%. The accuracy of Method B for mode-II is less than for mode-I, but acceptable results for most engineering applications, especially for geo-engineering applications, can be obtained even with coarse meshes. Severe errors are inevitable if the points where displacements are used for the calculation are very close to other fracture tips or boundaries of the computation domain. However, this is not unique to the GDC method, and other comparable methods suffer under the same conditions because the near-tip region is inadequately resolved. To correctly model these problems (e.g. tips close to each other or to the boundaries), sufficiently fine meshes must be adopted.

We found that the correction factor is a function of a number of variables for a give near-tip mesh configuration, including fracture length relative to element size, the Poisson's ratio, and random mesh perturbation. However, if we ignore these effects, by using correction factors derived for infinitely long fractures with a nominal Poisson's ratio of 0.2 on a regular mesh, the error is still likely to be within 10%.

Only the correction multipliers for quadratic six-node triangle elements are presented in this paper. Correction multipliers for any combination of element type and mesh configuration can be easily determined through a small number of FEM simulations following the procedure in Section 4. Only one crack-loading configuration needs be considered, and the resultant correction multipliers can be used in arbitrary fracture-load configurations with the same mesh.

Acknowledgments

This work was performed under the auspices of the US Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. The work of Fu and Carrigan in this paper was supported by the Geothermal Technologies Program of the US Department of Energy, and the work of Johnson and Settgast was supported by the LLNL LDRD project "Creating Optimal Fracture Networks" (#11-SI-006). This paper is LLNL report LLNL-JRNL-501931. We also would like to credit the two anonymous reviewers for advice leading to significant quality improvement of the paper.

References

[1] Tada H, Paris PC, Irwin GR. The stress analysis of cracks handbook. New York: ASME; 2000.
[2] Rice JR. A path independent integral and the approximate analysis of strain concentration by notches and cracks. J Appl Mech 1968;35:379–86.
[3] Parks DM. A stiffness derivative finite element technique for determination of crack tip stress intensity factors. Int J Fract 1974;10(4):487–502.
[4] Chan SK, Tuba IS, Wilson WK. On the finite element method in linear fracture mechanics. Engng Fract Mech 1970;2(1):1–17.
[5] Banks-Sills L, Sherman D. Comparison of methods for calculating stress intensity factors with quarter-point elements. Int J Fract 1986;32(2):127–40.
[6] Banks-Sills L, Einav O. On singular, nine-noded, distorted, isoparametric elements in linear elastic fracture mechanics. Comput Struct 1987;25(3):445–9.

[7] Zhu WX, Smith DJ. On the use of displacement extrapolation to obtain crack tip singular stresses and stress intensity factors. Engng Fract Mech 1995;51(3):391–400.
[8] Barsoum RS. On the use of isoparametric finite elements in linear fracture mechanics. Int J Numer Meth Engng 1976;10(1):25–37.
[9] Shih CF, deLorenzi H, German MD. Crack extension modeling with singular quadratic isoparametric elements. Int J Fract 1976;12(3):647–51.
[10] Tracey DM. Discussion of 'on the use of isoparametric finite elements in linear fracture mechanics' by R.S. Barsoum. Int J Numer Meth Engng 1977;11(2):401–2.
[11] Li FZ, Shih CF, Needleman A. A comparison of methods for calculating energy release rates. Engng Fract Mech 1985;21(2):405–21.
[12] Lim I, Johnston IW, Choi SK. On stress intensity factor computation from the quarter-point element displacements. Commun Appl Numer Meth 1992;8(5):291–300.
[13] Lim I, Johnston IW, Choi SK. Comparison between various displacement-based stress intensity factor computation techniques. Int J Fract 1992;58(3):193–210.
[14] Courtin S, Gardin C, Bezine G, Ben-Hadj-Hamouda H. Advantages of the *J*-integral approach for calculating stress intensity factors when using the commercial finite element software ABAQUS. Engng Fract Mech 2005;72(14):2174–85.
[15] Henshell RD, Shaw KG. Crack tip finite elements are unnecessary. Int J Numer Meth Engng 1975;9(3):495–507.
[16] Barsoum RS. Triangular quarter-point elements as elastic and perfectly-plastic crack tip elements. Int J Numer Meth Engng 1977;11(1):85–98.
[17] Ingraffea AR, Manu C. Stress-intensity factor computation in three dimensions with quarter-point elements. Int J Numer Meth Engng 1980;15(10):1427–45.
[18] Banks-Sills L, Bortman Y. Reappraisal of the quarter-point quadrilateral element in linear elastic fracture mechanics. Int J Fract 1984;25(3):169–80.
[19] Yehia NAB, Shephard MS. On the effect of quarter-point element size on fracture criteria. Int J Numer Meth Engng 1985;21(10):1911–24.
[20] Lynn PP, Ingraffea AR. Transition elements to be used with quarter-point crack-tip elements. Int J Numer Meth Engng 1978;12(6):1031–6.
[21] Banks-Sills L. Update: application of the finite element method to linear elastic fracture mechanics. Appl Mech Rev 2010;63(2):020803.
[22] Fu PC, Johnson SM, Carrigan CR. Simulating complex fracture systems in geothermal reservoirs using an explicitly coupled hydro-geomechanical model. In: Proceedings of the 45th US rock mechanics/geomechanics symposium, 11-244, San Francisco, CA, June 26–29; 2011.
[23] Guinea GV, Planas J, Elices M. K_I evaluation by the displacement extrapolation technique. Engng Fract Mech 2000;66(3):243–55.
[24] Williams MD, Jones R, Goldsmith GN. An introduction to fracture mechanics – theory and case studies. In: Transactions of mechanical engineering, vol. ME 14, IEAust, Australia, No. 4; 1989.
[25] Harrop LP. The optimum size of quarter-point crack tip element. Int J Numer Meth Engng 1982;18(7):1101–3.
[26] Srawley JE. Wide range stress intensity factor expressions for ASTM E 399 standard fracture toughness specimens. Int J Fract 1976;12(3):475–6.

INTERNATIONAL JOURNAL FOR NUMERICAL AND ANALYTICAL METHODS IN GEOMECHANICS
Int. J. Numer. Anal. Meth. Geomech. 2013; **37**:2278–2300
Published online 23 August 2012 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nag.2135

An explicitly coupled hydro-geomechanical model for simulating hydraulic fracturing in arbitrary discrete fracture networks

Pengcheng Fu*,†, Scott M. Johnson and Charles R. Carrigan

[1]*Atmospheric, Earth, and Energy Division, Lawrence Livermore National Laboratory, Livermore, CA 94550, U.S.A.*

SUMMARY

Modeling hydraulic fracturing in the presence of a natural fracture network is a challenging task, owing to the complex interactions between fluid, rock matrix, and rock interfaces, as well as the interactions between propagating fractures and existing natural interfaces. Understanding these complex interactions through numerical modeling is critical to the design of optimum stimulation strategies. In this paper, we present an explicitly integrated, fully coupled discrete-finite element approach for the simulation of hydraulic fracturing in arbitrary fracture networks. The individual physical processes involved in hydraulic fracturing are identified and addressed as separate modules: a finite element approach for geomechanics in the rock matrix, a finite volume approach for resolving hydrodynamics, a geomechanical joint model for interfacial resolution, and an adaptive remeshing module. The model is verified against the Khristianovich–Geertsma–DeKlerk closed-form solution for the propagation of a single hydraulic fracture and validated against laboratory testing results on the interaction between a propagating hydraulic fracture and an existing fracture. Preliminary results of simulating hydraulic fracturing in a natural fracture system consisting of multiple fractures are also presented. Copyright © 2012 John Wiley & Sons, Ltd.

Received 17 February 2012; Revised 25 June 2012; Accepted 10 July 2012

KEY WORDS: hydraulic fracture; discrete fracture network; explicit coupling; fracture interaction; rock joint; reservoir model

1. INTRODUCTION

Hydraulic fracturing is widely used by the energy industry (e.g., stimulation of gas shales, enhanced geothermal systems, etc.) to increase permeability of geological formations through the creation of hydraulically driven fractures and coupling of these new higher permeability flow paths with the natural fracture networks in the rock. A number of methods have been developed to make direct and indirect field observations on the hydraulic fracturing process, including mineback experiments, tiltmeter and microseismic mapping, pumping pressure diagnosis, and others. [1–4]. Numerous analytical and numerical hydraulic fracturing models have been developed to help interpret these observations (e.g., [5–10]). Despite the variety of existing models, there remains a gap between the state-of-the-art methodologies for modeling hydraulic fractures and the imminent needs of industry to improve prediction of hydraulically driven fracture behavior in the presence of complex preexisting fracture networks at field scales. Field data have demonstrated the complex patterns of new hydraulic fractures and remobilized preexisting fractures in naturally fractured reservoirs (e.g., [1, 11]). However, much attention from the hydraulic fracture modeling community has focused on scenarios with highly idealized fracture geometries. The classic Perkins–Kern–Nordgren and Khristianovich–Geertsma–DeKlerk (KGD) models [5, 7–9] and contemporary incarnations (e.g., [12, 13]) only address

*Correspondence to: Pengcheng Fu, Atmospheric, Earth, and Energy Division, Lawrence Livermore National Laboratory,7000 East Avenue, L-286, Livermore, CA 94550, U.S.A.
†E-mail: fu4@llnl.gov

propagation of a single fracture with assumed geometries in a homogeneous medium. The pseudo-3D (P3D) and planar 3D (PL3D) models [10, 14] are capable of addressing some of the issues with the homogeneous medium assumption by simulating fractures vertically extending through multiple geologic layers, but each simulation can only handle one crack lying in a single vertical plane. Other available numerical models for hydraulic fracturing generally approach modeling from one of two directions: rigorously address the solid–fluid coupling for a single fracture in a homogenous medium or address the relatively complex network, but with little or no ability to capture the creation of new fractures [15–21].

Here, we present a numerical method for simulating hydraulically driven fracturing in relatively complex preexisting/natural fracture systems, under the assumptions of quasi-static plane-strain deformation, laminar Newtonian flow in fractures, and an impermeable rock matrix. This numerical model is based on assumptions compatible with those for the KGD model, but the new model can handle arbitrary rock toughness and the interactions between multiple fractures. The organization of the paper is as follows: Section 2 of the paper describes the overall simulation strategy and the coupling scheme between the multiple physical processes involved. The algorithmic aspects of the individual components in the model are described in Section 3. In Sections 4 and 5, we verify and validate the model against available closed-form solutions for the propagation of a single fracture and laboratory experimental data on the interaction between two fractures, respectively. Finally, we present a preliminary example of the stimulation of a naturally fractured reservoir with an arbitrary preexisting fracture network.

2. STIMULATION STRATEGY AND COUPLING SCHEME

The aforementioned gap between existing simulation capabilities and the need for modeling arbitrary fracture systems is largely due to the intrinsically complex nature of the hydraulic fracturing process. A variety of inter-dependent physical mechanisms, including flow within the discrete fracture network in the presence of changing joint permeability, rock deformation caused by both interaction between the pressurized fluid in the joint and changing stresses within the rock matrix, and evolution of the fracture network and rock matrix topology as fractures propagate over time, must be appropriately handled to result in reasonable hydraulic fracturing simulation.

Existing analytical models often accommodate the interactions between the mentioned mechanisms by implicitly coupling them into the governing equations. Because of the complexity of the interactions, only a subset of the mechanisms, usually in highly simplified and idealized forms, can be incorporated into such equations. To avoid this limitation, our numerical model adopts an explicit coupling simulation strategy where individual modules are developed to model these distinct physical mechanisms with their interactions embodied by the data/information exchange between the modules. Because the solid and fluid solvers share the same time-integration approaches, the overall error from this approach of coupling remains second-order. Important modules in our numerical model include the following:

- A FEM geomechanics solver for linearly elastic solid and a linear elastic fracture mechanics (LEFM) component to resolve trajectory and growth rate of propagating fractures.
- A finite volume method (FVM) hydrodynamics flow solver for viscous, laminar flow.
- A geomechanical joint model to capture the nonlinear, hysteretic behavior of the interfacial interactions as well as the coupling to permeability changes.
- An adaptive remeshing module for generating topologically compatible meshes between the finite elements and finite volume elements.

Figure 1 illustrates the coupling of these modules. The algorithmic aspects are described in the next section. In real geological settings, a number of additional phenomena may be important depending on the application, including anisotropy, creep, plastic deformation, geochemical interactions, thermal effects, and others, which are also possible to treat by this numerical model with enhanced modules, but are beyond the scope of the present paper. The objective here is to develop a numerical model for hydraulic fracturing that can reasonably handle interactions between solid and fluid and those between fractures with a relatively rigorous treatment of fracture mechanics. The formulations of the constitutive modules in this paper serve this objective, and thus simple forms are preferred. Because

DOI: 10.1002/nag

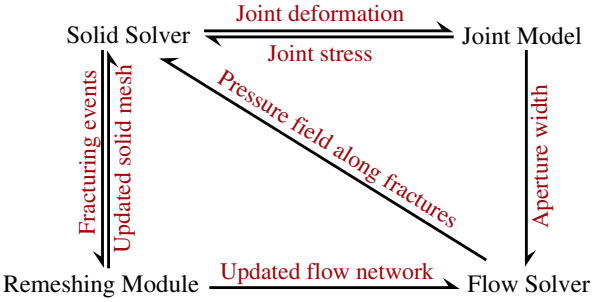


Figure 1. Important modules in the hydraulic fracturing simulator and their coupling.

of the modular design of the simulation framework and the explicit coupling method, each module can be easily modified or upgraded when necessary, as long as the interfaces with other modules are appropriately handled.

3. FORMULATION OF INDIVIDUAL MODULES

3.1. Finite element method solid solver

The finite element module uses the six-node isoparametric triangular plane-strain element known as the linear-strain triangle or the Veubeke triangle, and linear elasticity and small deformations are assumed for the intact material response. The solver uses a central-difference explicit time-integration scheme. At each time step, t, the nodal force vector $\mathbf{F}_i(t)$ acting on a node i has four contributions: (i) elastic deformation of the elements connected to the node; (ii) fluid pressure if the node is associated with a flow cell; (iii) contact stress if this node is associated with a closed joint; and (iv) external forces such as those acting at the stress-controlled boundaries. In the explicit time-integration scheme, the dynamic responses are solved on a nodal basis as follows.

$$\ddot{\mathbf{u}}_i(t) = \frac{\mathbf{F}_i(t) - \mathbf{F}_i^C(t)}{m_i} \tag{1}$$

$$\dot{\mathbf{u}}_i(t + \Delta t_s/2) = \dot{\mathbf{u}}_i(t - \Delta t_s/2) + \ddot{\mathbf{u}}_i(t)\Delta t_s \tag{2}$$

$$\mathbf{u}_i(t + \Delta t_s) = \mathbf{u}_i(t) + \dot{\mathbf{u}}_i(t + \Delta t_s/2)\Delta t_s \tag{3}$$

where $\mathbf{u}_i, \dot{\mathbf{u}}_i$, and $\ddot{\mathbf{u}}_i$ are the nodal displacement, velocity, and acceleration vectors, respectively. $\mathbf{F}_i^C$ is the nodal damping force, and only the mass-proportional term of the Rayleigh damping is used in this model. To reduce the computational constraints, the high frequency components in the dynamic response are filtered through the use of the damping term, which is commensurate with the quasi-static assumptions of the process. The mass of an element is distributed to the six nodes, with 1/19 of the element mass assigned to each vertex node and 16/57 to each mid-edge node (Section 16.2.4 in [22]). Δt_s is the time increment used in the solid solver and a CFL coefficient of 5% is used to ensure numerical stability. Typical hydraulic fracturing processes are quasi-static. The motivations for using a dynamic solver are to provide a robust solving scheme for this ill-conditioned problem and to enable a straightforward interface to couple with other modules.

3.2. Hydrodynamics solver for discrete flow network

Fluid flow in open rock fractures is idealized as laminar flow between two parallel plates employing lubrication theory. The governing equations used in typical hydraulic fracturing models are

DOI: 10.1002/nag

$$\frac{\partial q}{\partial l} + \frac{\partial w^h}{\partial t} = 0 \tag{4}$$

$$\kappa \frac{\partial P}{\partial l} = -q \tag{5}$$

$$\kappa = \frac{\left(w^h\right)^3}{12\mu_f} \tag{6}$$

where l represents the length along the fracture; q is the local flow rate in the fracture at a given cross-section; w^h is the local time-dependent hydraulic aperture size; P is the local fluid pressure; and κ represents the permeability of the fracture, which is a function of the dynamic viscosity μ_f of the fluid and the local aperture size w^h. Equation (4) is the continuity (mass conservation) equation; Equation (6) is the permeability equation, according to the laminar parallel plate flow assumption. These governing equations are solved with a two-dimensional FVM formulated on the basis of a three-dimensional approach described by Johnson and Morris [23]. This approach and the modifications are described here.

Implementations of FVM employ either node-centered (vertex-centered) or element-centered (cell-centered) formulations, and our model uses the latter. To avoid ambiguity, we use the nomenclature of 'cell' to denote a finite volume flow element and 'element' to denote a solid finite element. As shown in Figure 2, flow connections (corresponding to fracture networks in the solid phase) are discretized and visualized as line segments. For a given cell, i, the parameters correspond to length, L_{Ci}, fluid mass inside the cell, m_{Ci}, volume, V_{Ci}, hydraulic aperture size, w_i^h as well as the associated permeability, κ_{Ci} according to Equation (6), fluid pressure, P_{Ci}, and so on, where the subscript '$_C$' abbreviates 'cell'. In a cell representing an open fracture, the aperture size can be approximated by the distance between the two fracture walls calculated in the solid solver, and the volume is the product of length (area in 3D) of the cell along the direction of fracture extension, and the aperture size, that is, $V_{Ci} = L_{Ci} w_i^h$. The formulation for closed fractures subjected to compression will be discussed in Section 3.5. Fluid pressure and aperture size vary within each cell, so P_{Ci} represents the pressure value at the cell center, and w_i^h is the average aperture width of the cell.

The flow solver employs an explicit integration scheme, which makes it convenient to couple the flow solver and the solid solver. At each time step, the flow rate of the flow cells is evaluated on a node-by-node basis (note the distinction between the solid node and flow node). Assume there are N_I^C cells connected to the same flow node I; flow rate from a cell to the common node (i.e., outflow) is assumed to be negative; and the fluid pressure at this node is P_I. The flow rate between cell I-i (the ith cell connected to node I) and node I is

$$q_{I-i} = \frac{2\kappa_{I-i}(P_I - P_{I-i})}{L_{I-i}} \tag{7}$$

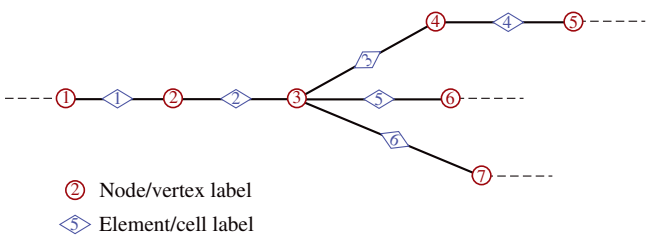


Figure 2. Two-dimensional flow network modeled by the finite volume method.

DOI: 10.1002/nag

To satisfy mass conservation, we have

$$\sum_{i=1}^{N_I^C} q_{I-i} = 0 \tag{8}$$

Combining Equations (7) and (8) yields

$$P_I = \frac{\sum P_{I-i}\kappa_{I-i}/L_{I-i}}{\sum \kappa_{I-i}/L_{I-i}} \tag{9}$$

and subsequently, the flow rate of each cell can be computed according to Equation (7). A special, yet very common case is that a flow node is connected to only two cells, denoted as cell i and cell j. In this particular case, the flow rate from cell i to cell j can be simplified as

$$q_{ij} = \frac{2\left(P_i - P_j\right)\kappa_{ij}}{L_i + L_j} \tag{10}$$

where κ_{ij} is the homogenized permeability with

$$\kappa_{ij} = \frac{\kappa_i \kappa_j \left(L_i + L_j\right)}{\kappa_i L_i + \kappa_j L_j} \tag{11}$$

By looping through all the flow nodes, we calculate the flow rate of each cell from and to its two nodes, thereby obtaining the mass increment and updated fluid mass in the cell. The local fluid pressure is related to the fluid density through the following equation-of-state (EOS)

$$P_{Ci} = \begin{cases} K_f\left(1 - \dfrac{\rho_{\text{ref}} V_{Ci}}{m_{Ci}}\right) & \text{if } m_{Ci}/V_{Ci} \geqslant \rho_{\text{ref}} \\ P_{\text{vap}} & \text{if } m_{Ci}/V_{Ci} < \rho_{\text{ref}} \end{cases} \tag{12}$$

where K_f is the bulk modulus of the fluid; ρ_{ref} is the reference density of this fluid, namely the density at zero or the datum pressure; P_{vap} is the temperature-dependent vapor pressure of this fluid, which is assumed to be zero, as the pumping pressure is typically many orders of magnitude higher than the vapor pressure. For any given fluid, the three parameters μ_f, K_f, and ρ_{ref} are dependent on temperature, and to a lesser extent, the pressure. These parameters are assumed to be constant in all the numerical examples of this paper unless otherwise indicated, but temperature-dependent and pressure-dependent material parameters can also be specified in the model. At the end of this step, fluid pressure is calculated for all the flow cells, and the procedure is repeated in successive time steps. The coupling between the solid phase deformation and the fluid flow is completed through the joint model, which applies the fluid pressure to the solid mesh elements that are interfaced with the cell and alters the aperture according to the geometric distance of the interfacing surfaces. Despite the simple form, this approach captures salient features of flow in narrow joints caused by a pressure gradient, and mass conservation and pressure variation in flow channels with constantly varying volume (i.e., varying aperture size). The second mechanism can cause discontinuity in the system and violate Equation (4), which is mediated through the EOS (12).

In this approach, fluid bulk modulus K_f acts as a component of the contact stiffness as well as of the EOS. The governing Equations (4)–(6) are essentially formulation for incompressible fluid, but K_f is used in the EOS to relate fluid density to pressure. The role of K_f in this solver is similar to that of material density in an explicit solid solver for quasi-static problems. That is, as a pseudo-inertial term, the fluid compressibility can be judiciously reduced to achieve a longer critical time step without sacrificing accuracy of the quasi-static analysis. We have empirically found that as long as the value of K_f is significantly greater than the fluid pressure, the simulation results are insensitive.

DOI: 10.1002/nag

3.3. *Fracturing criterion*

A fracturing criterion determines whether new fracture surface should be generated and along which direction the fracture should propagate by evaluating certain mechanical quantities at tips of existing fractures. The fracturing criterion in the current model is based on the 'critical stress intensity factor' concept in linear elastic fracture mechanics (LEFM). Mode-I and mode-II stress intensity factors (SIF), K_I and K_{II} are calculated using the generalized displacement correlation (GDC) method, and the propagation direction is determined using a maximum circumferential stress criterion for mixed-mode fracturing. Details of the GDC method and the evaluation of its accuracy are described in a separate paper [24], and we present the essence of this method here for completeness of the current paper.

3.3.1. Generalized displacement correlation method. In the original displacement correlation methods [25–30], SIF's are calculated from nodal displacements near the fracture tip based on analytical solutions for near-tip region displacement. It requires the use of special quarter-point elements [25, 28] in the first layer of elements around each tip, which makes it very difficult to be used in simulations of dynamic fracture propagations, where locations of fracture tips constantly evolve and are not known a priori. To overcome this problem, we have developed a generalized form of this method, called the GDC method, which uses regular linear or quadratic finite element types and can produce accurate results with relatively coarse mesh without near-tip refinement.

The finite element mesh near a fracture tip is shown in Figure 3. Quadratic elements (six-node triangle or eight-node quadrilateral) with mid-edge nodes are used. For plane-strain condition, SIF's can be calculated as

$$K_I = -\frac{\sqrt{\pi}C_I G\left(2u_\theta^A - 2u_\theta^{A'} - u_\theta^B + u_\theta^{B'}\right)}{2(1-\nu)\left(2-\sqrt{2}\right)\sqrt{l_E}} \tag{13}$$

$$K_{II} = -\frac{\sqrt{\pi}C_{II} G\left(2u_r^A - 2u_r^{A'} - u_r^B + u_r^{B'}\right)}{2(1-\nu)\left(2-\sqrt{2}\right)\sqrt{l_E}} \tag{14}$$

where G is the shear modulus of the solid; ν is the Poisson's ratio; l_E is the characteristic length of the element as denoted in Figure 3; C_I and C_{II} are correction multipliers; u_r and u_θ are the polar and angular displacements of reference points relative to the fracture tip. The GDC formulation is similar to that of the original displacement correlation method with the main difference being that the two reference points A and A' are mid-edge nodes of regular quadratic elements, instead of special quarter-point nodes. The multipliers C_I and C_{II} are necessary for correcting the errors induced by the inability of regular finite elements to characterize the square-root displacement term in the near-tip region. They were found in [24] to be functions of element type, tip-region mesh configuration, mesh size relative to crack length, and Poisson's ratio of the solid. Among these factors, the element type is fixed in

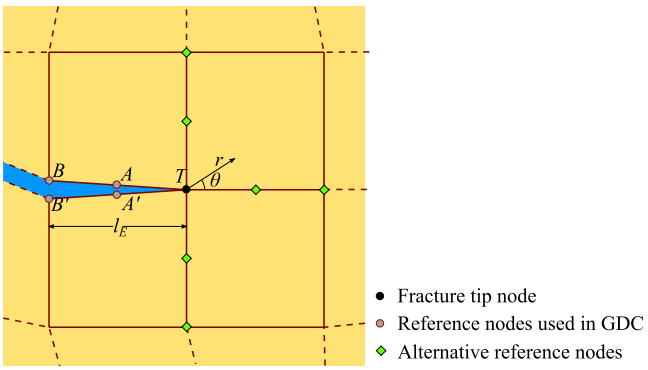


Figure 3. Typical mesh arrangement around a fracture tip. A polar coordinate system is established with its origin at the tip. The reference points used in Equations (13) and (14) are denoted as small circles, whereas alternative reference points shown as diamonds can also be used with modified formulations as elaborated in [24].

DOI: 10.1002/nag

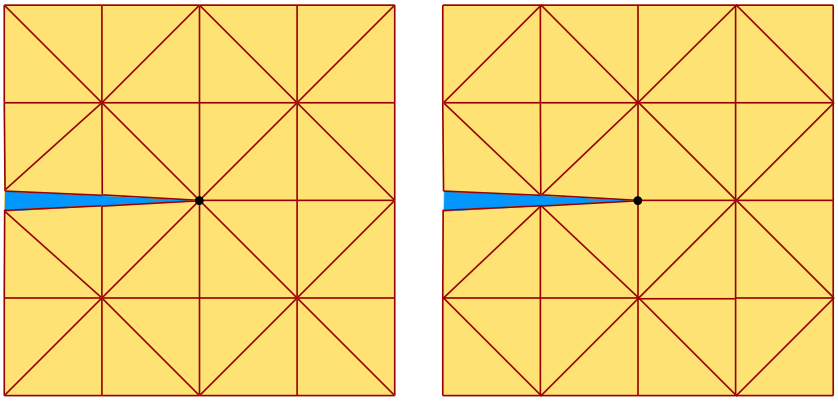
our model and the meshing scheme used in all the examples of the present paper consists of two common tip-region element configurations as shown in Figure 4. The significance of other factors is investigated in [24] and not repeated here. The correction multipliers for the meshing scenarios used in the present paper are shown in Figure 4.

3.3.2. Fracturing criterion and fracture propagation direction. When the SIF's at a tip are known, the well known fracturing criteria [31, 32] can be directly applied, and these criteria usually also predicts fracture propagation directions. In our current implementation of the numerical models, a simplified form of the criteria is adopted on the basis of the constraint that fracture trajectory can only follow element boundaries. Although it is theoretically possible to allow a fracture to propagate along an arbitrary direction through element partition or the extended finite element method (XFEM) [16], such methods will make the implementation of the model unacceptably expensive and complex, especially because of the coupling of multiple modules and the large number of possible scenarios of fracture intersections. The adopted meshing scheme shown in Figure 4 allows a fracture to propagate along seven or three directions (with 45° and 90° increments, respectively) from a tip. However, at a scale larger than the element size, a fracture can propagate along almost any direction by combining many element edges. Nevertheless, application of the fracturing criterion to be adopted needs only to determine along which candidate edge the fracture should propagate.

The simplified fracture criterion is triggered when $K_I^2 + K_{II}^2 \geqslant K_{\text{crit}}^2$ and $K_I > 0$, where K_{crit} is the critical stress intensity factor (i.e., toughness) of the matrix rock. The fracture toughness of rocks obtained from laboratory tests or the effective fracture toughness that takes tip-zone plasticity into consideration [33] may be used as K_{crit}. The second condition ($K_I > 0$) dictates that a fracture should not grow unless it is completely open. In the absence of pressurized fluid, all fractures in natural geological formations should be closed and compressive stress is transferred through the contact stress of the two walls of the fractures. As the fracture is pressurized with fluid, the two walls may slide as the normal contact stress, which is essentially the effective normal stress, decreases. Therefore, K_{II} may significantly develop before the fracture is open. However, because of the kinematical constraints posed by the closed fracture, fracture growth is bounded. This bounded sub-fracturing is homogenized in our model by only allowing fractures that are completely open to propagate.

Once the mentioned triggering criteria are met, we calculate the normal stress on all the candidate edges at their mid-edge nodes. The fracture will propagate along the edge with the greatest normal stress (tension is positive). The stress component resembles the circumferential stress used in some classical criteria (e.g., [31]), but it is evaluated at a distance of half the edge length instead of at the tip to be consistent with the approach of separating the entire edge during the time step.

This empirical criterion was found to yield reasonable results, as demonstrated by the numerical example in Section 6.1, where a hydraulic fracture propagates in a heterogeneous *in situ* stress field.



(a) Configuration a, C_I=1.5 and C_{II}=2.8 (b) Configuration b, C_I=1.9 and C_{II}=3.0

Figure 4. Two common tip-region mesh configurations used in this study and the corresponding GDC correction multipliers.

DOI: 10.1002/nag

3.4. Adaptive remeshing algorithm

When the fracturing criterion determines that a fracture should grow along an identified finite element edge, this edge is flagged 'fracture-ready' and the adaptive remeshing module is invoked. We use the idealized example shown in Figure 5 to demonstrate how the solid mesh and flow cells are updated. The current remeshing module only handles the so-called 'node-split' in order to create new fracture faces. More sophisticated scenarios, such as adaptive mesh refinement [34] can be accommodated in this framework when necessary, but are not described here.

An edge is considered to be external if there is only one element attached to it, whereas there are always two and only two elements attached to an internal edge. An external edge either represents the free boundary of the rock mass, or one of the walls along a fracture interface. Each time a fracture-ready edge is identified, the two nodes attached to this edge are examined. A given node will be split if either two of the edges connected to this node are fracture-ready or one of the edges connected to this node is fracture-ready while two of the edges are external. Figure 5(a) illustrates this approach with edges 8 and 13 flagged as fracture-ready. Subsequently, node 5 is split from the first condition and nodes 4 and 9 are split owing to the second condition. Figure 5(b) shows the mesh configuration after the aforementioned remeshing has taken place. Each node that has been split generated two daughter nodes. For instance, nodes 12 and 13 are the daughter nodes of node 5. The daughter nodes belong to the new solid mesh whereas the mother nodes are detached from the solid mesh and attached to the newly created flow cells (cell 1 and cell 2). Reusing the nodes and edges that have been detached from the solid mesh ensures that intersecting fractures will result in correct connectivity of the new flow cells. For instance, edge 5 is flagged at a later step, and subsequently, nodes 12 and 2 are split. The new flow cell 3 should not be connected to node 12, which has just been split, but to node 5, the mother node of node 12 as shown in Figure 5(c). During the remeshing process, the mapping between mother nodes and daughter nodes, and that between mother edges and daughter edges is established and stored. Such information is used frequently during the simulation because we apply the fluid pressure from flow cells (which were all previously solid element edges) to their daughter edges as stress boundary conditions to the solid solver. Meanwhile, the locations of the daughter nodes and daughter edges are used to update the locations of the flow cells and the aperture sizes.

3.5. Joint model

Joint behaviors involved in a hydraulic fracturing process, such as dilation associated with shear deformation, reversed and cyclic loading, joint asperity degradation, and their influences on hydraulic conductivity are very complex, and sophisticated constitutive models are often needed to deal with these behaviors (e.g., [35–37]). Here, we have implemented a simplified form of the joint model that

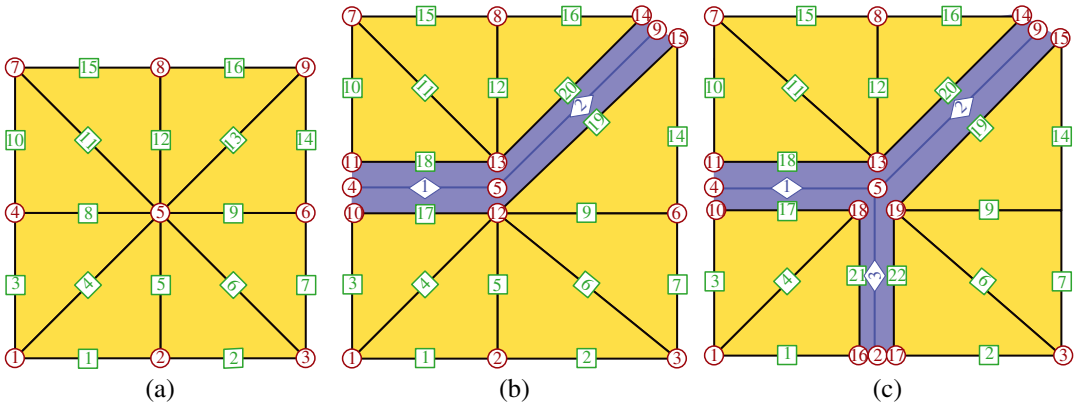


Figure 5. Adaptive remeshing of the finite element model to create new fractures. (a), (b), and (c) represent three states of the same mesh. The labels for edges are placed at mid-edge, and the mid-edge nodes are not shown. Because of the unique correspondence between the edges and the mid-edge nodes, mid-edge nodes are always split when the corresponding edge is split.

DOI: 10.1002/nag

handles the most basic behaviors including the opening, closing, shear deformation and sliding. As illustrated in Figure 1, the essential function of the joint model in this numerical model is to receive information regarding rock deformation from the solid solver, calculate stress responses and permeability changes, and feed this information to the solid and flow solvers.

Figure 6 shows two solid elements on the opposing sides of a fracture. Two edges, denoted as edge p and edge q, of these elements represent the two opposing walls along the fracture. Edge p is geometrically characterized by its mid-point $\mathbf{x}^p$ in the vector form, its length L^p, a unit outer-pointing normal vector $\mathbf{n}^p$, and a unit tangential vector $\mathbf{t}^p$. Similar variables can be defined for edge q and are not repeated here. These two edges are the daughter edges of the same edge in the original non-fractured solid mesh, so the lengths are the same ($L^p = L^q$) if the small difference in deformation of the two elements in the tangential direction is ignored. The two edges are assumed to be parallel, that is, $\mathbf{n}^p + \mathbf{n}^q \approx \mathbf{0}$. The normal and tangential components of the distance between the mid-points of the two edges are

$$\delta_n = (\mathbf{x}^q - \mathbf{x}^p)\cdot\mathbf{n}^p \tag{15}$$

$$\delta_t = (\mathbf{x}^q - \mathbf{x}^p)\cdot\mathbf{t}^p \tag{16}$$

The rate of change of the previously mentioned quantities, $\dot{\delta}_n$ and $\dot{\delta}_t$ can be calculated using similar formulations, but with the location vectors replaced with velocity vectors. Because the relative displacement of the two sister edges in the tangential direction in hydraulic fracturing simulations is usually very small compared with the length of the edges, only contacts between sister edges are evaluated. In other words, it is possible that a very small segment of edge p can interact with a segment of the edge next to edge q along the fracture face, but this type of interaction is ignored in the model, and no neighbor-sorting is performed to update nearest neighbors, which both limits and expedites the calculation.

When $\delta_n < 0$, the two elements that these two edges attach to penetrate into each other geometrically. This small virtual penetration is used as a 'penalty' term in the finite element solver, and it represents the state that the two walls along the fracture are in contact. Contact stresses are generated as a function of the virtual penetration, in a fashion similar to how contact mechanics is handled in the discrete element method. The absolute value of δ_n is conveniently equivalent to the joint closure as used in rock mechanics. The normal contact stress and the tangential contact stress are calculated using the following equations:

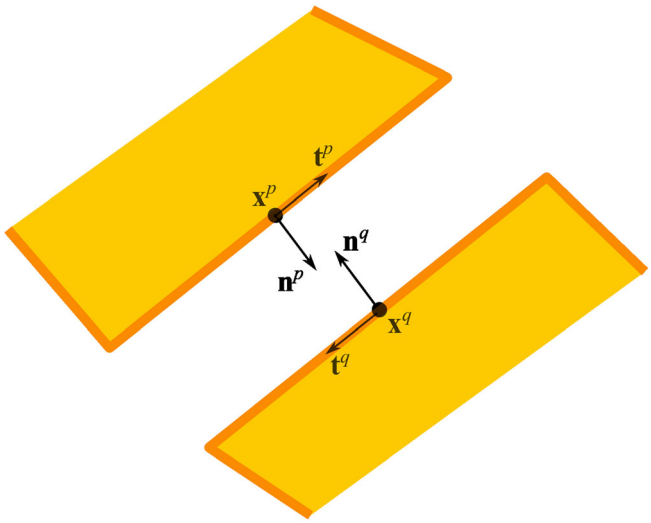


Figure 6. Geometrical characterization of two opposing edges along a fracture. The distances between the two edges are exaggerated for illustration purposes.

DOI: 10.1002/nag

$$(\sigma_n)_{t+\Delta t} = (\sigma_n)_t + k_n \dot{\delta}_n \Delta t \tag{17}$$

$$\tau_{t+\Delta t} = \begin{cases} \tau_t + k_s \dot{\delta}_t \Delta t & \text{if } \left|\tau_t + k_s \dot{\delta}_t \Delta t\right| < \left|(\sigma_n)_{t+\Delta t}\right| \mu_J \\ \text{sign}(\tau_t)\left|(\sigma_n)_{t+\Delta t}\right| \mu_J & \text{otherwise} \end{cases} \tag{18}$$

where k_n and k_s are the normal stiffness and shear stiffness of the joint with a dimension of stress/length. It is well known that both k_n and k_s are highly nonlinear, and they are strongly correlated. The responses of the systems simulated by the present model are not sensitive to joint stiffness, so k_n and k_s are treated as constants for illustration purposes. μ_J is the coefficient of friction of the walls along the fracture; sign() is a function returning the sign (positive or negative) of the argument. The Coulomb failure criterion is enforced through Equation (18). Note that although the omission of shear dilation is appropriate for the scope of the present paper, this mechanism can play a significant role in other problems. The combination of the contact stress and the fluid pressure should be applied to the edges along the fracture as stress boundary conditions.

When $\delta_n > 0$, we term it the mechanical aperture (also termed the storage aperture in the literature) of the fracture. As mentioned in Section 3.2, it is assumed that the permeability of an open fracture ($\delta_n > 0$) obeys the cubic law expressed in Equation (6), that is, $w^h = \delta_n$. When a fracture is closed ($\delta_n < 0$), it can still conduct fluid flow because of the partly continuous void space between the two walls left by the imperfect matching of the asperities on the opposing sides. Under this condition, the permeability is a function of many factors, including roughness and strength of the joint walls, the mismatch of the two walls, effective compressive normal stress, and shear dilation. These factors are not considered in the examples in this paper, and we instead use the following simplified relationship between the equivalent hydraulic aperture size w^h and δ_n:

$$w^h = \begin{cases} \delta_n & \text{if } \delta_n > w_0^h \\ w_0^h & \text{otherwise} \end{cases} \tag{19}$$

where w_0^h is the 'residual' equivalent hydraulic aperture size of a closed fracture, and it is assumed to be constant regardless of the closure and stress of the joint. Note that the 'equivalent' aperture size of a closed fracture is calculated from the permeability of the fracture according to Equation (6). It does not represent the physical opening of the two walls in contact, but it conveniently has a dimension of length.

4. MODEL VERIFICATION AGAINST THE KGD MODEL

4.1. The KGD model and its compatibility with the proposed numerical model

The KGD hydraulic fracture model was independently developed by Khristianovic and Zhelton [5], and Geertsma and de Klerk [8]. Because it is based on assumptions that are compatible with those of the proposed numerical model in this paper, we use the KGD model as the reference for model verification.

The KGD model concerns the propagation of a single fracture driven by fluid pumped into the fracture from the wellbore at a constant flow rate of q_0, as shown in Figure 7. It assumes plane-strain deformation, linearly elastic, homogeneous and isotropic media, and laminar Newtonian flow obeying the cubic law, which are consistent with the proposed numerical model. The KGD model assumes that the flow rate everywhere along the fracture is the same as that at the wellbore when calculating pressure loss. This simplification is not needed in the numerical model, and the effects of this assumption are discussed in Section 4.2. The propagation of fracture is controlled by the assumption that there is no gap (vacuum) between the front of the fluid and the fracture tip. Therefore, the KGD model essentially assumes the fracture propagation is in the fluid viscosity-controlled regime, and the rock toughness is not explicitly considered. The setup of the numerical model can adopt the same assumption as described in Section 4.2, but a more general case with finite rock toughness will be discussed in Section 4.3.

DOI: 10.1002/nag

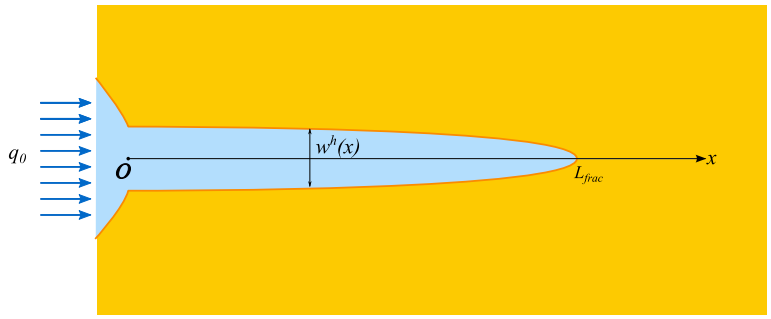


Figure 7. Geometrical characteristics of a KGD fracture. The well is partly shown. The model implies that there is another identical fracture on the other side of the well symmetrical to the one shown.

Pressurized fluid drives the fracture to propagate along the direction perpendicular to the minimum compressive principal stress. Closed-form solutions for various quantities, such as fracture length L_{frac} and aperture width at the wellbore w_0^h at any given time t are available, such as the set derived by Valko and Economides [38]:

$$L_{\text{frac}}(t) = 1.078\left(\frac{E' q_0^3}{\mu_f}\right)^{1/6} t^{2/3} \tag{20}$$

$$w_0^h(t) = 2.36\left(\frac{\mu_f q_0^3}{E'}\right) t^{1/3} \tag{21}$$

where $E' = 2\,G/(1-v)$ is the plane-strain modulus of elasticity.

4.2. *Numerical realization of the KGD model*

The simulated domain has dimensions of 100 and 120 m in the x and y directions, respectively and is discretized into 24,000 elements with a mesh pattern shown in Figure 4. The core mesh is then extended to approximately 1000 m in each dimension with progressively larger elements to mitigate boundary effects. Slip boundary conditions are applied to the edges. At the left side boundary, where the injection well is located, this applies as a symmetrical condition, consistent with the assumption in the KGD model. Because of the linear elasticity assumptions of the model, the *in situ* stress applied at the boundary will not affect the net pressure results, consistent with the KGD model. Simulation parameters used in this and subsequent analyses (whenever applicable) described in the paper are listed in Table I.

To realize the zero-toughness and no-vacuum assumptions of the KGD model, the critical SIF (K_{crit}) in the model is a small finite value (1000 Pa m$^{1/2}$). The fracturing criterion is only checked when the flow cell connected to the tip is fully filled with fluid, that is, $m_{Ci}/V_{Ci} \geqslant \rho_{\text{ref}}$. When a new flow cell

Table I. Parameters of the numerical model for the simulation of the KGD model.

Parameters	Value
Rock, shear modulus G	8.3 GPa
Rock, Poisson's ratio v	0.2
Fluid, dynamic viscosity μ_f	0.001 Pa s
Fluid, bulk modulus K_f	2.2 GPa
Flow rate at wellbore q_0	1.0, 2.0[a], and 4.0 L/s per meter thickness of reservoir
Residual hydraulic aperture width w_0^h	0.02 mm

Note:
[a]baseline case simulation.

DOI: 10.1002/nag

and the associated fracture are created, a fluid lag exists. Meanwhile, the aperture and volume of this cell continue to grow along with permeability. The fracture will start propagating from the current tip when the cell is fully filled and the calculated K_I is greater than the threshold value.

The growth of the fracture length calculated using the numerical model for the three injection rates listed in Table I is compared with the corresponding KGD analytical solutions in Figure 8. For the baseline case ($q_0 = 2.0$ L/s per meter reservoir thickness), three snapshots of aperture width along the fracture at $t = 20$, 40, and 80 s are shown in Figure 9. The KGD model assumes the cross section of the fracture to have an elliptical shape whereas the numerical model calculates the aperture width based on deformation in the solid phase, and no such assumption is needed. At each time, the integral of the aperture width along the fracture is the total volume of the fracture, which is the product of the injection flow rate and the injection duration. Compared with the analytical solution, the numerical model predicts a slightly shorter fracture length and slightly wider aperture at the well. Because a number of approximations had to be made in the derivation of the KGD solution, which can be relaxed in the numerical model (e.g., constant flow rate along the fracture), the small differences do not necessarily indicate error of the numerical model.

4.3. *Toughness-dominated regime*

The propagation of fracture in the original KGD model is dominated by viscous flow of the fluid, and a key assumption is that there is no gap (i.e., vacuum) between the front of fluid and the fracture tip. In this section, we derive the formulation for hydraulic fracturing in rocks with a high critical stress intensity factor and compare the numerical model with the analytical solution. We assume in this regime that aperture of the fracture is wide enough so that pressure loss of the fluid along the fracture is negligible compared with the fluid pressure at the tip, so the net pressure ΔP is a constant in the fracture from the wellbore to the fracture tip. The validity of this assumption in the numerical examples is established later in this section. Two wings of fractures grow simultaneously from the well towards opposite directions and their combination can be modeled as a planar fracture in an infinite medium with its center at the wellbore. Assuming at time t the length of each wing is $L_{\text{frac}}(t)$, the volume of fluid in one wing is

$$q_0 t = \frac{\Delta P \pi L_{\text{frac}}^2}{E'} \tag{22}$$

where E' is the plane-strain modulus of elasticity defined in Section 4.1, and q_0 is the fluid injection rate into each wing, which is a constant for a simulation. The net pressure ΔP is determined by the condition that the mode-I stress intensity factor at the tip equals to the rock toughness K_{crit}, namely

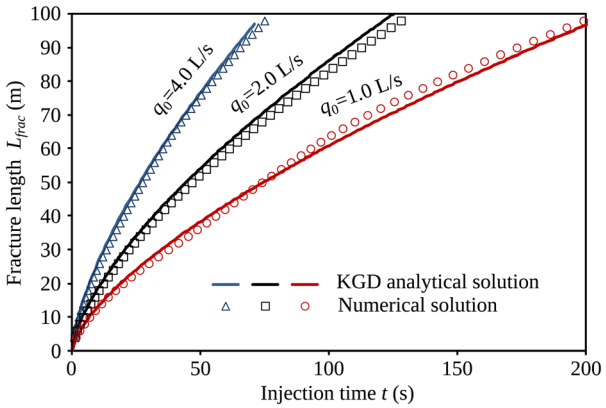


Figure 8. Comparison between the numerical model and the KGD analytical solution in terms of fracture growth rate.

DOI: 10.1002/nag

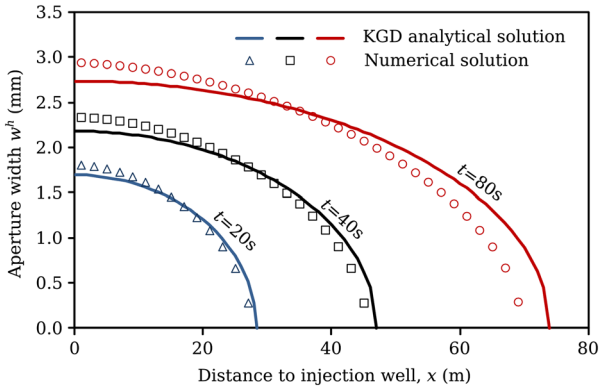


Figure 9. Comparison between the numerical model and the KGD analytical solution in terms of aperture distribution along the fracture.

$$\Delta P(\pi L_{\text{frac}})^{1/2} = K_{\text{crit}} \tag{23}$$

Plugging Equation (22) into (23), we can obtain

$$L_{\text{frac}}(t) = \pi^{-1/3}\left(\frac{q_0 E' t}{K_{\text{crit}}}\right)^{2/3} \tag{24}$$

Under this finite-toughness condition, the length of the fracture is proportional to the injection time raised to the exponent of 2/3, similar to the original KGD model shown in Equation (20). However, the viscosity of the fluid does not influence the fracture growth rate, but the rock toughness does. To numerically model this finite-toughness condition, the same numerical model described in Section 4.2 is adopted with the no-vacuum-at-tip restriction removed. Two cases with rock toughness values of 5.0 and 10.0 MPa m$^{1/2}$ are simulated, and the flow rate q_0 is assumed to be 2.0 L/s per meter thickness of reservoir. As shown in Figure 10, the numerical results match the analytical solution reasonably well.

To check the assumption that the pressure loss along the fracture can be ignored in these cases, we examine the following situation. For the case with $K_{\text{crit}} = 5.0$ MPa m$^{1/2}$ at $t = 75.2$ s, the fracture length L_{frac} is approximately 50 m. The mean aperture size is 3.0 mm ($\bar{w}^h = q_0 t / L_{\text{frac}}$). For a constant flow rate of $q_0 = 2.0$ L/s per meter reservoir thickness, through a fracture with a uniform aperture width of 3.0 mm, the pressure drop is 44 kPa according to Equations (5) and (6), which is approximately 11%

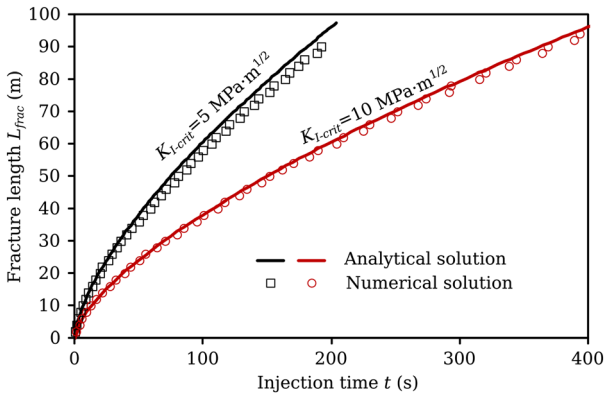


Figure 10. Comparison between the analytical solution and the numerical simulation results for the finite-toughness scenarios.

DOI: 10.1002/nag

of the net pressure required to create a stress intensity factor of 5.0 MPa m$^{1/2}$ at the fracture tip. For the case with $K_{I\text{-crit}} = 10.0$ MPa m$^{1/2}$ at a crack length of 50 m, this ratio is 0.7%. Therefore, omission of pressure loss along the fracture for the rock toughness-dominated scenarios is reasonable for the parameters tested. Note that the critical SIF values used in these two examples are higher than those of typical rocks, in order to ensure that the fracture propagation is in the toughness-dominated regime. Real hydraulic fracture propagation should be somewhere between these viscosity-dominated and toughness-dominated bounds. Although this is naturally accommodated by the proposed numerical model, analytical solutions for these intermediate scenarios are not available.

5. MODEL VALIDATION AGAINST LABORATORY TESTS

5.1. *Description of the laboratory test*

In the previous section, we have verified that the numerical model can appropriately handle the coupling between the fluid phase and the solid phase during the propagation of a single fracture. We further validate the model in this section in terms of its ability to simulate the interaction between a propagating fracture that intersects an existing fracture.

Blanton [39] fabricated synthetic rock blocks using 'hydrostone' with an existing fracture embedded at variable angles with respect to the specimen, as illustrated by the horizontal cross-section in Figure 11. Each rock block was then placed in a triaxial cell for testing. The vertical compressive stress (out-of-plane in Figure 11; compression is positive) is 20 MPa, and the two horizontal principal stress components $\sigma_h < \sigma_H \leqslant 20$ MPa. Water was injected into a hole at the center of each specimen to create a hydraulic fracture propagating in the plane normal to the minor principal stress σ_h and subsequently intersecting the existing fracture at an angle of approach θ_{apr}. The variables investigated in Blanton's study included the magnitudes of σ_H and σ_h and the angle of approach θ_{apr}. Testing parameters and the observed interaction modes between the hydraulic fracture and the existing fracture for selected cases are listed in Table II.

5.2. *Mechanisms for different interaction modes*

Three modes of interaction including 'crossing', 'arrest', and 'opening' were reported in Blanton's study. The mechanisms behind these three modes have been extensively studied using a variety of

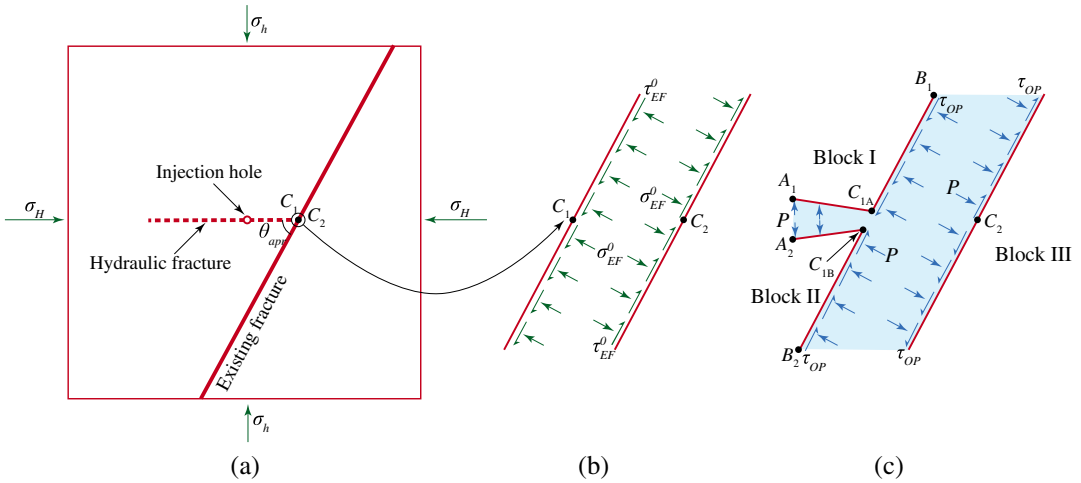


Figure 11. Schematic views of Blanton's laboratory tests [39] on the interaction between a hydraulic fracture and an existing fracture. The two horizontal dimensions of each specimen are 30 × 30 cm^2. (a) Triaxial stress applied to the specimen and the geometrical configuration of the two fractures; (b) stress along the existing fracture before the hydraulic fracture intersects it, and (c) *additional* stress along the existing fracture induced by the opening of the hydraulic fracture and the interaction between the two fractures.

DOI: 10.1002/nag

Table II. Different scenarios tested in Blanton (1982) [39], observed interaction modes, and numerical simulation results.

Case ID[a]	θ_{apr} (°)	Principal stress (MPa)		Interaction mode	Stress on ex. frac. before pumping		$\frac{\tau^0_{EF}}{\sigma^0_{EF}-\sigma_h}$	Max. SIF at C_2 (MPa m$^{1/2}$)
		σ_H	σ_h		σ^0_{EF}	τ^0_{EF}		
CT-20	90	14.0	5.0	Crossing	14.0	0.0	0.00	0.69
CT-21	60	14.0	5.0	Arrest	11.7	3.9	0.58	0.36
CT-8	60	20.0	5.0	Crossing	16.2	6.5	0.58	0.59
CT-4	60	12.0	10.0	Opening	11.5	0.9	0.58	0.13
CT-22	45	10.0	5.0	Opening	7.5	2.5	1.00	Negligible
CT-14	45	14.0	5.0	Arrest	9.5	4.5	1.00	Negligible
CT-13	45	16.0	5.0	Arrest	10.5	5.5	1.00	Negligible
CT-12	45	18.0	5.0	Arrest	11.5	6.5	1.00	Negligible
CT-11	45	20.0	5.0	Arrest	12.5	7.5	1.00	Negligible

Note:
[a]Case ID here is the 'Test #' in Blanton's original paper.

methods (e.g., [40–43]) in the literature. We briefly review the process of a hydraulic fracture intersecting an existing fracture to help the determination of the key parameters of the numerical model and the interpretation of the simulation results.

Before the stress field in the specimen is significantly altered by the creation and propagation of the hydraulic fracture, the normal and shear stress (σ^0_{EF} and τ^0_{EF}, respectively; the subscript '$_{EF}$' stands for 'Existing Fracture' and the superscript '^{0}' indicates that this is the initial condition) as shown in Figure 11(b) are functions of σ_H , σ_h , and θ_{apr} and are listed in Table II. When the hydraulic fracture has intersected the existing fracture at point C_1, but has not break the other wall of the fracture at C_2, additional stresses will act on the existing fracture: First, pressurized fluid will start to flow into the existing fracture. If we assume the fluid pressure at a given moment and location along the existing fracture is $P(t)$ and the effective normal stress (i.e., contact normal stress between the two walls along the fracture) at the same time and location is $\sigma_{EF}(t)$, the following relationship holds

$$\sigma_{EF}(t) + P(t) \approx \sigma^0_{EF}$$

which is approximate because the stress field might be perturbed near the intersection. The effective normal stress along the existing fracture decreases as the fluid pressure in it increases. Because solid block *I* (enclosed by A_1-C_{1A}-B_1) and block II (A_2-C_{1B}-B_2) tend to move away from each other, especially when fluid pressure P is significantly higher than σ_h, this motion will create additional shear stress τ_{OP} along the fracture as shown in Figure 11(c). Note that the subscript '$_{OP}$' stands for 'opening'. This shear stress increment has opposite directions at the two sides of point C_2, and is the primary driving mechanism of the stress intensity factor at C_2.

The role of hydraulic pressure in this process is twofold. To generate an SIF that is great enough to break the fracture wall at point C_2 and allow the hydraulic fracture to cross the existing fracture, high fluid pressure is needed to create additional shear stress τ_{OP} along the existing fracture by pushing blocks *I* and *II* apart. However, a higher fluid pressure will reduce the effective normal stress on the existing fracture, and the two blocks might be able to slide along the wall, preventing the creations of a high SIF. The relative significance of these mechanisms depends on the existing normal and shear stresses on the fracture before fluid flow into the existing fracture. Next, we consider an idealized scenario. Assume when the hydraulic fracture breaks one of the fracture walls at point C_1 and intersects the existing fracture, the fluid pressure $P = \sigma_h$. Note that $P \geqslant \sigma_h$ is the necessary condition (but no sufficient condition) for the hydraulic fracture to propagate. Because the hydraulic pressure merely balances σ_h, blocks *I* and *II* do not have a significant tendency to move apart from the fracture and therefore $\tau_{OP} \approx 0$. If we assume no sliding takes place along the existing fracture under this condition, the mobilized coefficient of friction is $\tau^0_{EF}/\left(\sigma^0_{EF} - P\right) = \tau^0_{EF}/\left(\sigma^0_{EF} - \sigma_h\right) = \tan\left(90^\circ - \theta_{\text{apr}}\right)$ with the derivation process omitted here and the value of mobilized coefficient of friction for all the scenarios

DOI: 10.1002/nag

listed in Table II. Under this condition, configurations with smaller approaching angles have a stronger tendency to slide along the existing fracture at the moment the hydraulic fracture intersects the existing fracture. If the fluid pressure near the intersection point increases beyond σ_h, then the sliding tendency is enhanced, because of the following:(i) the effective normal stress is further reduced; and (ii) the shear stress is increased at least on one side of point C_2.

5.3. Numerical simulation results

The nine scenarios in Table II are simulated using the numerical model. The original paper [39] did not provide information on the fluid pressure for each case. The actual pressure should be dependent on a number of factors, including the minor principal stress σ_h, dynamic response of the pumping system, and the compliance of the hydraulic system. A clue that helps estimate the pressure at the injection hole is the difference between the 'opening' mode and the 'arrest' mode. Opening means that the hydraulic fracture is first arrested by the existing fracture and the pumping pressure is higher than normal stress ($\sim \sigma^0_{EF}$) induced by the boundary condition. We found that a fluid pressure of $\sigma_h + 3.0$ MPa is consistent with the observations in the laboratory testing results, and this pressure is used as the flow boundary condition in the simulations. The toughness (i.e., critical stress intensity factor) of the hydrostone is unknown. To quantify the effects of the external variables on the ability or potential of the hydraulic fracture to cross the existing fracture, we use a small toughness value ($10\,\text{kPa}\,\text{m}^{1/2}$) on the left-hand-side of the existing fracture, so that the hydraulic fracture can propagate towards the existing fracture. We do not allow the mesh to fracture to the right of the existing fracture in the simulation. Instead, we track the mode-I stress intensity factor K_I at point C_2 and the maximum value achieved by each specimen is presented in Table II. Other numerical simulation parameters are presented in Table III.

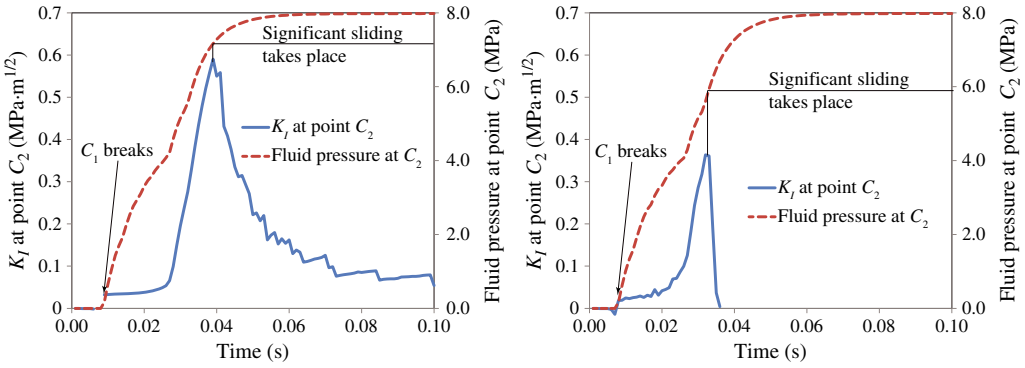
Figure 12 shows the evolution of K_I at point C_2 and the fluid pressure near C_2 for two cases, CT-8 and CT-21, with their only difference being σ_H. After the fracture wall at C_1 breaks, K_I at C_1 increases as the pressure increases, due to the associated increase of τ_{OP}. K_I suddenly drops when the pressure is high enough to allow sliding to occur along the existing fracture. The case with a higher σ_H value has a stronger resistance to sliding than the other case, and therefore K_I is able to continue to grow to a higher peak value before the fluid pressure is high enough to induce sliding. As mentioned in Section 5.3 and shown in Table II, for the specimens with $\theta_{\text{apr}} = 45°$, the mobilized coefficient of friction at C_2 needs to be higher than 1.0 to sustain significant τ_{OP}, but the coefficient of friction used in the simulation is 0.7. Therefore, K_I values significantly higher than zero cannot develop in those cases.

On the basis of the simulation results, we find that if the toughness of the hydrostone is greater than 0.36 but smaller than $0.59\,\text{MPa}\,\text{m}^{1/2}$, the numerical model can exactly reproduce the observed phenomena in Blanton's laboratory tests. Although the toughness of this particular material used cannot be precisely determined, the study in this section demonstrates that the proposed explicit coupling simulation strategy and the numerical model can adequately reflect the physical mechanisms governing the interaction between two intersecting fractures. A significant advantage of

Table III. Parameters of the numerical model for the simulation of the Blanton experiments.

Parameters	Value
Rock, shear modulus G	8.3 GPa
Rock, Poisson's ratio v	0.2
Fluid, dynamic viscosity μ_f	0.001 Pa s
Fluid, bulk modulus K_f	2.2 GPa
Fluid pressure at the injection hole	$\sigma_h + 3.0$ MPa
Joint, residual hydraulic aperture width w_0^h	0.005 mm
Joint, coefficient of friction μ_J	0.7
Joint, normal stiffness k_n	500 GPa/m
Joint, shear stiffness k_s	1.0 GPa/m
Average element dimension	~1 cm

DOI: 10.1002/nag



(a) Case CT-8, σ_H=20 MPa (b) Case CT-21, σ_H=14 MPa

Figure 12. The evolution of K_I at point C_2 and fluid pressure near C_2 for two cases.

this method compared with other methods used for this problem is that the temporal evolution of the states of each phase can be explicitly resolved and the effects of each variable in this physical process can be studied independently.

6. DEMONSTRATION OF SIMULATION CAPABILITY

6.1. *Fracture propagation in heterogeneous field*

The proposed numerical model only allows fracture to initiate and propagate along interfaces between neighboring solid elements, namely edges in the mesh, which raises mesh dependency concerns. In this section, we examine this effect through a numerical example where a hydraulic fracture propagates in a solid medium with a heterogeneous stress field.

The boundary conditions applied to a $200 \times 100\,\mathrm{m}^2$ solid medium are shown in Figure 13(a). Slip boundaries are applied at the left and bottom boundaries. The stress applied at the other two boundaries is denoted in the figure, and the resultant nodal stress tensors are visualized as ellipses. The major principal stress is horizontal at the left side of the medium, and it gradually becomes vertical at the right side. Fluid is pumped into the domain through a perforation shown in the figure. Because hydraulic fracture tends to grow in the direction perpendicular to the minor principal stress, it is expected that it will first propagate horizontally and then gradually turn vertical.

The simulation uses parameters similar to those used in Section 4, and the fracture path obtained and the stress tensor distribution at the end of the simulation are shown in Figure 13(b). Although the simulated hydraulic fracture abruptly switches trajectory by 45° due to the mesh constraints, the model is able to capture the overall propagation path of the fracture, which is dictated by the applied boundary conditions. Therefore, the mesh dependency of fracture path appears to be not a serious issue at a scale that is significantly larger than the element size. The fracturing criterion is 'smart' enough to find the optimum combination of element edges to form a continuous fracture path that is consistent with the mechanical conditions applied in the model.

6.2. *Study of responses of a reservoir with isotropically oriented natural fractures*

In this section, we use the proposed numerical model to investigate the stimulation of a virtual reservoir with the presence of largely isolated natural fractures with uniformly distributed orientations. The variables to be studied include the orientation of the far-field principal stress axes and the degree of stress anisotropy. The reservoir setting is hypothetical, and the primary objective of these simulations is to study whether the numerical simulation results can reasonably respond to the variation of external variables.

DOI: 10.1002/nag

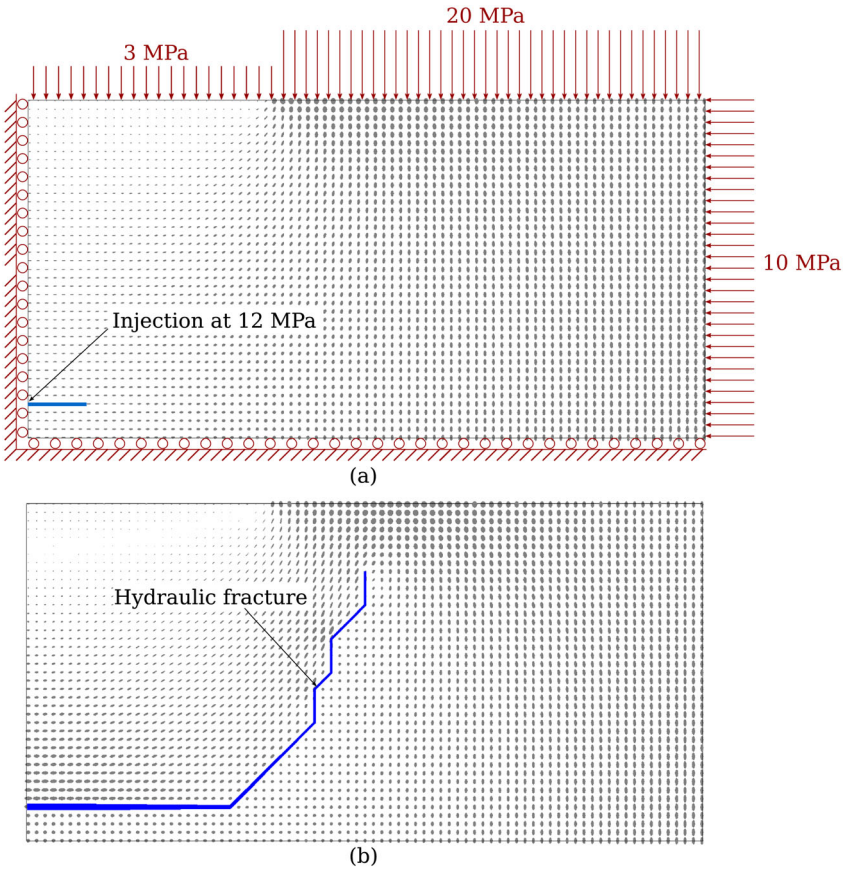


Figure 13. Hydraulic fracture propagation in a medium with a heterogeneous stress distribution. (a) The boundary conditions and stress tensor distribution before hydraulic fracturing. (b) The path of the hydraulic fracture and the stress tensor at the end of the simulation. The 2D stress tensor at each note is represented by an ellipse. The lengths of the two axes of an ellipse are proportional to the two principal stress components at this point, and the orientations of the two ellipse axes coincide with the orientations of the two principal stress components.

6.2.1. Natural fractures and meshing strategy. The simulation domain interior to the boundary mesh is 100 m long in both dimensions (from 0 to 100 m in the *x*/horizontal direction and from −50 to 50 m in the *y*/vertical direction), and the triangle elements have edges approximately 1 m long. The mesh is based on the meshing scheme shown in Figure 4, but a small and random perturbation is imposed on the location of each node to introduce some randomness to the mesh as shown in Figure 14(c). Progressively larger element sizes are employed to extend the simulation domain to 1000 m in each dimension, and the far-field stress conditions are applied at the boundary of the extended mesh. A preexisting natural fracture system is randomly generated and mapped onto the edges of solid elements within the core simulation domain as shown in Figure 14 (a). The fractures are largely isolated with lengths ranging from 6 to 18 m with a mean of 11 m. The orientations of these fractures are uniformly distributed between 0 and 180° rotating from the x direction. The injection well for hydraulic stimulation is placed at $x=0$ and $y=0$. At the bottom, top, and right boundaries of the core simulation domain, a zero-pressure boundary condition is specified in the flow solver as shown in Figure 14 (b), so these three boundaries are treated as fluid 'sinks'. Simulation parameters used in this suite of examples are similar to those used in Sections 4 and 5 and are thus not repeated here.

6.2.2. The effects of principal stress orientation. Three simulations are performed in the study of the effects of principal stress orientation. In the baseline case (A-1), the far-field stress is $\sigma_{xx}=15$ MPa, $\sigma_{yy}=10$ MPa, and $\sigma_{xy}=0$ (compressive stress is positive in this example). Fluid is pumped into the system through the injection well denoted in Figure 14(b) at a constant pressure of 14 MPa. The

DOI: 10.1002/nag

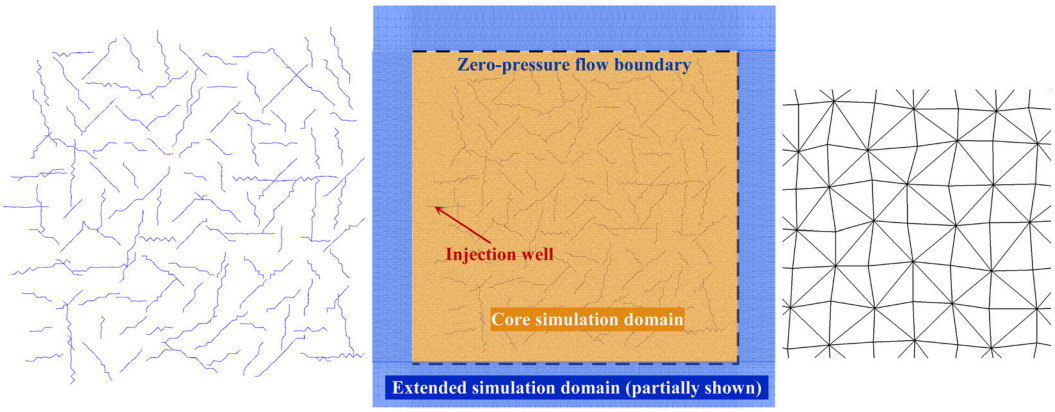


Figure 14. Preexisting natural fractures and the meshing strategy. (a) The randomly generated natural fractures; (b) the core simulation domain and the extended domain; and (c) perturbed mesh to introduce randomness to the fracture path.

simulation result of the stimulated fracture system for the baseline case at the end of the stimulation are shown in Figure 15(a), where the fractures (including both natural and created fractures) that are engaged (i.e., connected to the injection well and pressurized by the fluid) in the stimulation are shown in red color, and the unaffected fractures are in gray. Note that the aperture widths are magnified by 20 times to enable clear visualization. The distribution of stress component σ_{yy} at the end of the stimulation is shown in Figure 15(b) where the additional compressive stress created by the pressurized fractures along the horizontal direction and the tensile stress at fracture tips in the pumping front are visible. The simulation results for the two additions cases, A-2 and A-3, where the principal stresses have rotated counterclockwise and clockwise, respectively, by 30° are shown in Figure 15(c) and (d).

It is well known that hydraulic fractures tend to propagate along the plane of the least compressive (far-field) stress in homogeneous media. In all the three cases, the general orientations of the engaged fracture systems are consistent with the predicted directions based on the far-field principal stress orientation. The heterogeneity in the rock body because of the presence of natural fractures inevitably affects the paths along which hydraulic fractures propagate, making them deviate from the ideally predicted paths. These effects appear to be local, with a minimal influence on the general trends of the fractures. Moreover, these effects, embodied by the interactions between fractures are well reflected in the numerical model. This study also confirms the observations made in Section 6.1 regarding the minimal mesh dependency of fracture paths in the proposed numerical method. In the current meshing scheme, the inter-element interfaces, namely potential fracture paths are generally along directions 0°, 45°, 90°, and 135° from the x-axis with some randomness introduced by the mesh perturbation. However, this does not prevent the fractures from propagating along directions $\pm 30°$.

6.2.3. The effects of stress anisotropy. In this study, the baseline case B-1 is the same as the baseline case A-1 in the previous study. The additional scenarios have the same far-field stress in the y-direction ($\sigma_{yy} = 10$ MPa) as the baseline case, but smaller compressive stresses $\sigma_{xx} = 12$ and 10 MPa for cases B-2 and B-3, respectively. Note that the far-field stress for case B-3 is isotropic. The pumping pressure for all the cases remains 14 MPa. The simulation results are shown in Figure 16 in a fashion similar to that of Figure 15, with fractures engaged in the stimulation highlighted. The result for B-1 is the same as that for A-1, and is thus not repeated in Figure 16.

The stimulated fracture network for case B-2 is similar to that of the baseline case, with a slightly more diffuse pattern of fracture growth at the far side from the injection well, presumably because the reduced compressive stress in the x-direction provides more flexibility in the choice of viable propagation paths by the hydraulic fracture. In case B-3, the isotropic far-field *in situ* stress does not

DOI: 10.1002/nag

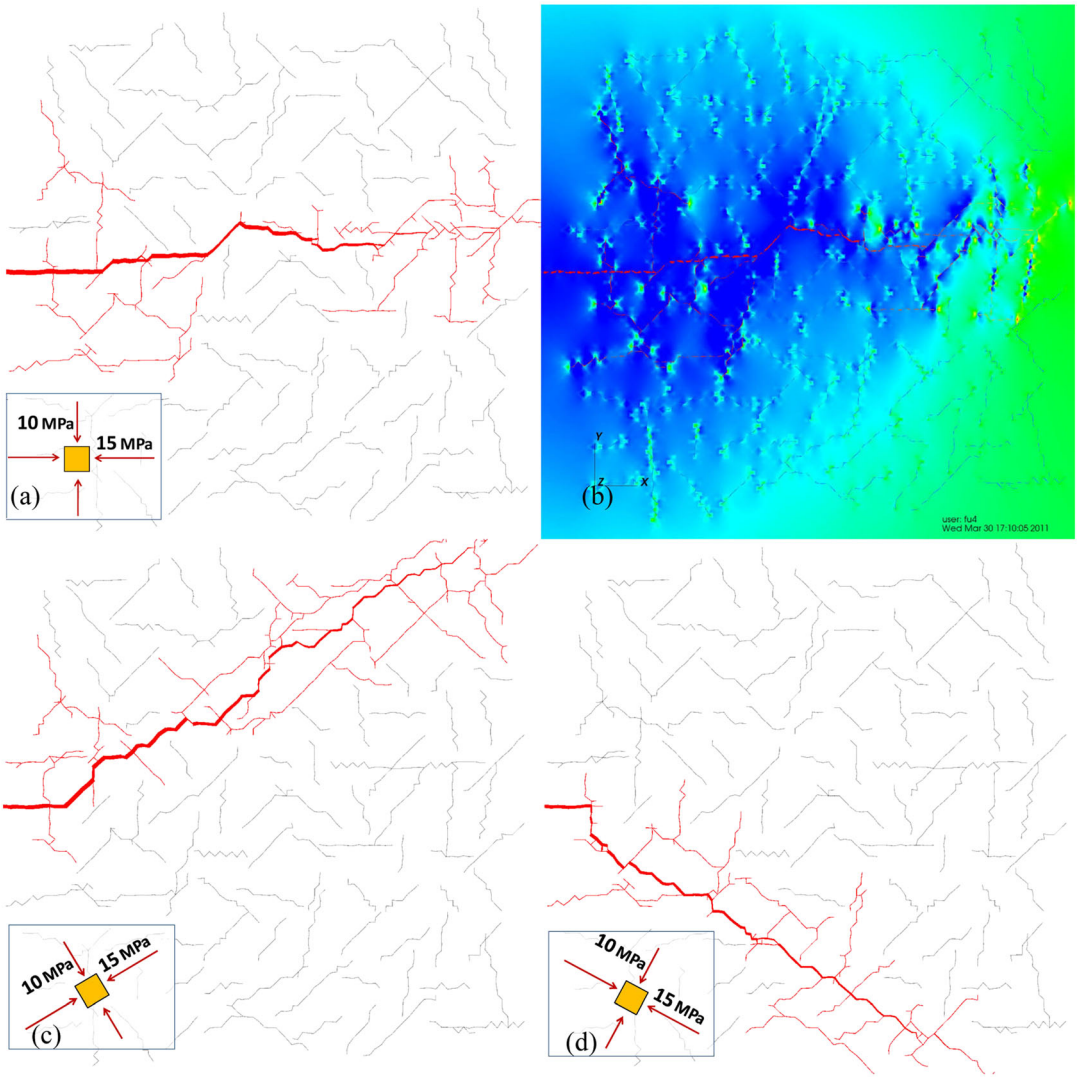


Figure 15. Stimulated fracture networks with different far-field principal stress orientations. Preexisting natural fractures and newly created fractures that are engaged by the stimulation are shown in red color, whereas unaffected natural fractures are in gray. The orientations of the principal stresses are schematically shown in each figure. (a) Baseline case A-1 where the major principal stress aligns with the x-axis; (b) distribution of σ_{yy} in case A-1 at the end of stimulation, with the blue end of the color spectrum indicating stress that is more compressive and the red end being more tensile or less compressive; (c) case A-2, the principal stress axes have rotated counterclockwise by 30° from the baseline; and (d) case A-3, the principal stress axes have rotated clockwise by 30° from the baseline.

impose a preferential fracture propagation direction. Four major branches of fractures have developed as the results of the stimulation along largely random directions, but these four branches tend to propagate away from each other. This is because if two parallel fractures are close to each other, the compressive stress in the rock matrix induced by the fluid pressure tends to impede the development of tensile zones at the fracture tips, retarding further propagation. One may argue that these four branches appear to propagate generally along the element edge directions. This phenomenon indicates that the mesh configuration plays a secondary role in determining the fracture propagation direction. Although *in situ* stress and existing fractures are more significant factors, as revealed by other examples, the subtle role of mesh configuration may become visible when the effects of these primary factors vanish under certain conditions (e.g., isotropic stress field and isotropic fracture orientation).

DOI: 10.1002/nag

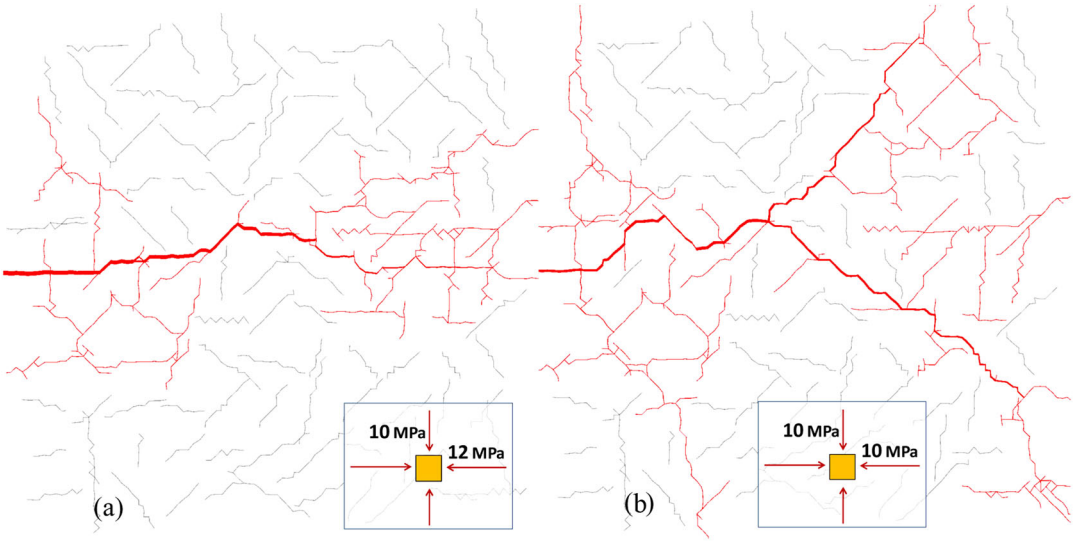


Figure 16. Stimulated fracture networks under different degrees of stress anisotropy, but the same principal stress axis orientation. The far-field stress state is denoted in each figure. Note that the baseline case is shown in Figure 15(a).

7. CONCLUDING REMARKS

In this paper, we present an explicit coupling simulation strategy for hydraulic fracturing in arbitrary natural fracture systems. In the proposed method, each of the physical processes involved in hydraulic fracturing is modeled by a separate module of the simulator and the interactions between these processes are embodied by the data/information sharing between these modules. Because multiple processes of rather different natures are involved, and they influence each other in many different ways, this explicit strategy provides a flexible simulation framework for complex hydraulic fracturing phenomena. Because the operations of these modules are sufficiently decoupled in this explicit coupling methodology, upgrading an individual module with more realistic (and inevitably more complex) models can be done without significantly affecting other modules. Therefore, although only the baseline simulation capability using relatively simple models is described in this paper, the presented simulation framework will remain unchanged if more complex problems are to be simulated.

The verification and validation of the numerical model focus on relatively simple but well-quantifiable phenomena in rock-fracture-fluid systems. Quantitative data, at either the laboratory or field scale are not available for the interaction between hydraulic fractures and existing natural fracture networks. However, because the interactions between fractures in a complex fracture network can be decomposed into the propagation of individual fractures and the interactions within individual pairs of fractures, the verification and validation in this paper provide a reasonable physical and mechanics base, on which the credibility of the proposed model is built.

The numerical model using this explicit coupling strategy is known to be computationally expensive. The main reason is that the physical phenomena being simulated are not only complex, but also ill-conditioned. The simulation domain is often hundreds of meters in each direction whereas typical aperture width is a small fraction of a millimeter. A deformation that is considered small ‘noise’ in the solid solver may induce dramatic (by orders of magnitudes) oscillation of fluid pressure in the flow solver. Because the model has to essentially resolve multiple dynamic physical processes with characteristic length-scales across several orders of magnitude, the time steps used in both the solid and flow solvers are necessarily very small. As an example, each of the simulations in Section 6.2 costs hundreds of CPU-hours on currently mainstream computers. A number of modeling and computational technologies, including more efficient solvers, more intelligent time-stepping, hybrid solvers, and massively parallelized processing are currently being developed and implemented to enable this model to be used more effectively.

DOI: 10.1002/nag

The initial development of the proposed numerical method was on the platform of Livermore Distinct Element Code (LDEC). LDEC is a massively parallel multi-physics simulator developed by the Computational Geosciences Group at the Lawrence Livermore National Laboratory to simulate the response of jointed geologic media to dynamic loading. Additional capabilities, including combined FEM-DEM analysis, fracture mechanics, and explicit solid–fluid coupling have been implemented in LDEC in the continued development over the past decade [23, 44–46]. However, the methodology described in this paper is portable and can be implemented in any appropriate numerical platform.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Geothermal Technologies Program of the US Department of Energy for support of this work under the Enhanced Geothermal Systems Program. Additional support was provided by the Lawrence Livermore National Laboratory (LLNL) LDRD project 'Creating Optimal Fracture Networks' (#11-SI-006). The authors also would like to acknowledge their collaborators at the LLNL. This work was performed under the auspices of the US Department of Energy by LLNL under Contract DE-AC52-07NA27344. This paper is LLNL Report LLNL-JRNL-519272.

REFERENCES

1. Warpinski NR, Teufel LW. Influence of geologic discontinuities on hydraulic fracture propagation. *Journal of Petroleum Technology* 1987; **39**(2):209–220. doi:10.2118/13224-PA.
2. Nolte KG, Smith MB. Interpretation of fracturing pressures. *Journal of Petroleum Technology* 1981; **33**(9):1767–1775. doi:10.2118/8297-PA.
3. Warpinski NR, Branagan PT, Peterson RE, Wolhart SL, Uhl JE. Mapping hydraulic fracture growth and geometry using microseismic events detected by a wireline retrievable accelerometer array. *Proceedings of SPE Gas Technology Symposium*, 1998. Society of Petroleum Engineers. DOI: 10.2118/40014-MS.
4. Cipolla CL, Wright CA. Diagnostic techniques to understand hydraulic fracturing: what, why, and how. *Proceedings of SPE/CERI Gas Technology Symposium*, 2000. Society of Petroleum Engineers. DOI: 10.2118/59735-MS.
5. Khristianovic SA, Zheltov YP. Formation of vertical fractures by means of highly viscous liquid. *Proceedings of the Fourth World Petroleum Congress*, Rome, 1955; 579–586.
6. Hubbert KM, Willis DG. Mechanics of hydraulic fracturing. *Transactions of The American Institute of Mining and Metallurgical Engineers* 1957; **210**(6):153–163.
7. Perkins TK, Kern LR. Widths of hydraulic fractures. *Journal of Petroleum Technology* 1961; **13**:937–949.
8. Geertsma J, de Klerk F. A rapid method of predicting width and extent of hydraulically induced fractures. *Journal of Petroleum Technology* 1969; **21**:1571–1581.
9. Nordgren RP. Propagation of a vertical hydraulic fracture. *Society of Petroleum Engineers Journal* 1972; **12**:306–314.
10. Adachi J, Siebrits E, Peirce A, Desroches J. Computer simulation of hydraulic fractures. *International Journal of Rock Mechanics and Mining Sciences* 2007; **44**:739–757. doi:10.1016/j.ijrmms.2006.11.006.
11. King GE, Haile L, Shuss J, Dobkins TA. Increasing fracture path complexity and controlling downward fracture growth in the Barnett shale. *Proceedings of SPE Shale Gas Production Conference* (SPE 119896), 2008. DOI: 10.2118/119896-MS.
12. Detournay E. Propagation regimes of fluid-driven fractures in impermeable rocks. *International Journal of Geomechanics* 2004; **4**(1):35–45. doi:10.1061/(ASCE)1532-3641(2004)4:1(35).
13. Garagash DI, Detournay E. Plane-strain propagation of a fluid-driven fracture: small toughness solution. *Journal of Applied Mechanics* 2005; **72**(6):916–928. doi:10.1115/1.2047596.
14. Siebrits E, Peirce AP. An efficient multi-layer planar 3D fracture growth algorithm using a fixed mesh approach. *International Journal for Numerical Methods in Engineering* 2002; **53**(3):691–717. doi:10.1002/nme.308.
15. Galindo Torres SA, Muñoz Castaño JD. Simulation of the hydraulic fracture process in two dimensions using a discrete element method. *Physical Review E* 2007; **75**(6):1–9. doi:10.1103/PhysRevE.75.066109.
16. Dahi-Taleghani A. Analysis of hydraulic fracture propagation in fractured reservoirs: an improved model for the interaction between induced and natural fractures. Doctoral dissertation, Texas A&M University, 2009.
17. Chen Z, Bungera AP, Zhang X, Jeffrey RG. Cohesive zone finite element-based modeling of hydraulic fractures. *Acta Mechanica Solida Sinica* 2009; **22**(5):443–452. doi:10.1016/S0894-9166(09)60295-0.
18. Sarris E, Papanastasiou P. The influence of the cohesive process zone in hydraulic fracturing modelling. *International Journal of Fracture* 2010; **167**(1):33–45. doi:10.1007/s10704-010-9515-4.
19. Damjanac B, Gil I, Pierce M, *et al*. A new approach to hydraulic fracturing modeling in naturally fractured reservoirs. *Proceedings of the 44th US Rock Mechanics Symposium and 5th US-Canada Rock Mechanics Symposium*, Salt Lake City, Utah, ARMA 10–400, 2010.
20. Zhang Z, Ghassemi A. Simulation of hydraulic fracture propagation near a natural fracture using virtual multidimensional internal bonds. *International Journal for Numerical and Analytical Methods in Geomechanics* 2011; **35**(4):480–495. doi:10.1002/nag.905.

21. Hunsweck MJ, Shen Y, Lew AJ. A finite element approach to the simulation of hydraulic fractures with lag. *International Journal for Numerical and Analytical Methods in Geomechanics* 2012. doi:10.1002/nag.1131.
22. Zienkiewicz OC, Taylor RL, Zhu JZ. *Finite Element Method – Its Basis and Fundamentals*. Elsevier Butterworth-Heinemann: Oxford, 2005.
23. Johnson SM, Morris JP. Modeling hydraulic fracturing for carbon sequestration applications. *Proceedings of the 43rd US Rock Mechanics Symposium and the 4th US-Canada Rock Mechanics Symposium*, Asheville, NC, ARMA 09–30, 2009.
24. Fu P, Johnson SM, Settgast RR, Carrigan CR. Generalized displacement correlation method for estimating stress intensity factors. *Engineering Fracture Mechanics* 2012; **88**:90–107.
25. Barsoum RS. On the use of isoparametric finite elements in linear fracture mechanics. *International Journal for Numerical Methods in Engineering* 1976; **10**(1):25–37.
26. Shih CF, de Lorenzi H, German MD. Crack extension modeling with singular quadratic isoparametric elements. *International Journal of Fracture* 1976; **12**(3):647–651.
27. Tracey DM. Discussion of 'on the use of isoparametric finite elements in linear fracture mechanics' by R. S. Barsoum. *International Journal for Numerical Methods in Engineering* 1977; **11**(2):401–402.
28. Barsoum RS. Triangular quarter-point elements as elastic and perfectly plastic crack tip elements. *International Journal for Numerical Methods in Engineering* 1977; **11**(1):85–98.
29. Ingraffea AR, Manu C. Stress-intensity factor computation in three dimensions with quarter-point elements. *International Journal for Numerical Methods in Engineering* 1980; **15**(10):1427–1445.
30. Banks-Sills L, Sherman D. Comparison of methods for calculating stress intensity factors with quarter-point elements. *International Journal of Fracture* 1986; **32**(2):127–140.
31. Erdogan F, Sih GC. On the crack extension in plates under plane loading and transverse shear. *Journal of Basic Engineering* 1962; **85**(4):519–527. doi:10.1115/1.3656897.
32. Sih GC. Strain-energy-density factor applied to mixed mode crack problems. *International Journal of Fracture* 1974; **10**(3):305–321. doi:10.1007/BF00035493.
33. Papanastasiou P. The effective fracture toughness in hydraulic fracturing. *International Journal of Fracture* 1999; **96**(2):127–147.
34. Schrefler B, Secchi S, Simoni L. On adaptive refinement techniques in multi-field problems including cohesive fracture. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**(4–6):444–461.
35. Barton N, Bandis S, Bakhtar K. Strength, deformation and conductivity coupling of rock joints. *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts* 1985; **22**(3):121–140. doi:10.1016/0148-9062(85)93227-9.
36. Cook NGW. Natural joints in rock: mechanical, hydraulic and seismic behaviour and properties under normal stress. *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts* 1992; **29**(3):198–223. doi:10.1016/0148-9062(92)93656-5.
37. Yeo IW, de Freitas MH, Zimmerman RW. Effect of shear displacement on the aperture and permeability of a rock fracture. *International Journal of Rock Mechanics and Mining Sciences* 1998; **35**(8):1051–1070. doi:10.1016/S0148-9062(98)00165-X.
38. Valko P, Economides MJ. *Hydraulic Fracture Mechanics*. Wiley: New York, 1995.
39. Blanton TL. An experimental study of interaction between hydraulically induced and pre-existing fractures. *Proceedings of SPE Unconventional Gas Recovery Symposium. Pittsburgh, Pennsylvania*. Society of Petroleum Engineers, 1982; 559–571. DOI: 10.2118/10847-MS.
40. Renshaw CE, Pollard DD. An experimentally verified criterion for propagation across unbounded frictional interfaces in brittle, linear elastic materials. *International Journal of Rock Mechanics and Mining Science & Geomechanics Abstracts* 1995; **32**(3):237–249. doi:10.1016/0148-9062(94)00037-4.
41. Zhang X, Jeffrey RG. Reinitiation or termination of fluid-driven fractures at frictional bedding interfaces. *Journal of Geophysical Research* 2008; **113**(B8):1–16. doi:10.1029/2007JB005327.
42. Akulich AC, Zvyagin AV. Interaction between hydraulic and natural fractures. *Fluid Dynamics* 2008; **43**(3):428–435. doi:10.1134/S0015462808030101.
43. Gu H, Weng X. Criterion for fractures crossing frictional interfaces at non-orthogonal angles. *Proceedings of the 44th US Rock Mechanics Symposium and 5th U.S.-Canada Rock Mechanics Symposium*, Salt Lake City, Utah, ARMA 10–198, 2010.
44. Morris JP, Rubin MB, Block GI, Bonner MP. Simulations of fracture and fragmentation of geologic materials using combined FEM/DEM analysis. *International Journal of Impact Engineering* 2006; **33**:463–473. doi:10.1016/j.ijimpeng.2006.09.006.
45. Block G, Rubin MB, Morris J, Berryman JG. Simulations of dynamic crack propagation in brittle materials using nodal cohesive forces and continuum damage mechanics in the distinct element code LDEC. *International Journal of Fracture* 2007; **144**:131–147. doi:10.1007/s10704-007-9085-2.
46. Morris JP, Johnson SM. Dynamic simulations of geological materials using combined FEM/DEM/SPH analysis. *Geomechanics and Geoengineering: An International Journal* 2009; **4**:91–101. doi:10.1080/17486020902767354.

PROCEEDINGS, Thirty-Seventh Workshop on Geothermal Reservoir Engineering
Stanford University, Stanford, California, January 30 - February 1, 2012
SGP-TR-194

USING FULLY COUPLED HYDRO-GEOMECHANICAL NUMERICAL TEST BED TO STUDY RESERVOIR STIMULATION WITH LOW HYDRAULIC PRESSURE

Pengcheng Fu, Scott M. Johnson, and Charles R. Carrigan

Atmospheric, Earth, and Energy Division, Lawrence Livermore National Laboratory
7000 East Ave., L-286
Livermore, CA 94550, USA
e-mail: fu4@llnl.gov

ABSTRACT

This paper documents our effort to use a fully coupled hydro-geomechanical numerical test bed to study using low hydraulic pressure to stimulate geothermal reservoirs with existing fracture network. In this low pressure stimulation strategy, fluid pressure is lower than the minimum *in situ* compressive stress, so the fractures are not completely open but permeability improvement can be achieved through shear dilation. The potential advantage of low pressure stimulation compared with high pressure stimulation is that a large fracture network instead of a single primary fracture can be stimulated. We found that in this low pressure regime, the coupling between the fluid phase and the rock solid phase becomes very simple, and the numerical model can achieve a low computational cost. Using this modified model, we study the behavior of a single fracture and a random fracture network.

INTRODUCTION

Geological formations in rocks with low initial permeability can be hydraulically stimulated to create enhanced (or engineered) geothermal reservoirs with enhanced permeability and thereby improved heat production efficiency (MIT study, 2006). The conceptual model for hydraulic stimulation that is most commonly referred to depicts the following process. When fluid pressure exceeds the minimum principal stress in the rock formation, new hydraulic fractures initiate and propagate along the plane that is perpendicular to the minimum principal stress direction. These new hydraulic fractures will intersect existing natural fractures in the formation and form a interconnected fracture network, through which fluid in the production phase can flow from the injection well to the production well(s) and bring heat from the hot rocks covered by this network.

A concern over the process described in this conceptual model is that once a hydraulic fracture (termed primary fracture herein) is opened, the conductivity along this fracture from the injection point to the fracture front is much higher than the neighboring fractures that are still closed. Meanwhile, the high fluid pressure in this open fracture creates a "stress shadow" around this fracture, which increases the rock matrix compressive stress experienced by neighbor fractures. The direct consequence of these two effects is that this open fracture will continue to grow at a relatively high rate, thereby further strengthening these effects, whereas the neighbor closed fractures may never be able to open and subsequently compete with the primary fracture. This is true regardless whether the primary fracture is a newly created hydraulic fracture or an existing fracture that happens to be oriented normal to the minimum principal stress. In this scenario, only one fracture (the primary fracture) can be stimulated. Even though it is possible to obtain high permeability between the injection well and the production well through this primary fracture, heat in a small volume in the reservoir around this fracture can be harvested, which is highly undesired for enhanced geothermal system (EGS) stimulation. It is possible to stimulate multiple fractures and create interconnected fracture network using technologies such as horizontal drilling with staged fracking, but such technologies are expensive and more applicable to shale gas production than to EGS.

An alternative stimulation strategy is to stimulate the reservoir at a fluid pressure lower than the minimum principal stress in the rock matrix. No new fractures will be created and none of the existing fractures will be completely open. In this scenario, instead of stimulate a single fracture, the fracture network which must already be interconnected prior to the stimulation will be stimulated by "hydro-shearing" (Willis-Richards et al., 1996). This paper investigates the mechanisms of low pressure hydraulic

stimulation using a fully coupled hydro-geomechanical numerical test bed developed at the Lawrence Livermore National Laboratory. The numerical algorithms in this numerical test bed, which originally focuses on high-pressure hydraulic fracturing, have been documented elsewhere (Johnson and Morris, 2009; Fu et al., 2011). In this paper, we describe the modifications to the original algorithms that enable high-efficiency simulation of low pressure stimulation in this paper, as well as various numerical examples on low pressure stimulation.

STRESS SHADOWING CONSIDERATION

First, we quantitatively evaluate the evolution of stress shadowing, namely the increase of rock matrix stress as fluid pressure in a fracture increases. Consider an infinite array of parallel fractures with infinite length as a highly idealized scenario that enables a closed-form solution to be obtained, as illustrated in Figure 1. The distance between any two neighboring fractures is H. Initially the fluid pressure in these fractures is P_F=0 and the rock matrix stress normal to the fractures is σ_{Mi}. As the fractures are simultaneously pressurized with fluid, they will begin to dilate and the rock matrix stress σ_M will increase accordingly. The effective normal stress along these fractures

$$\sigma'_J = \begin{cases} \sigma_M - P_F & \text{if } \sigma_M > P_F \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

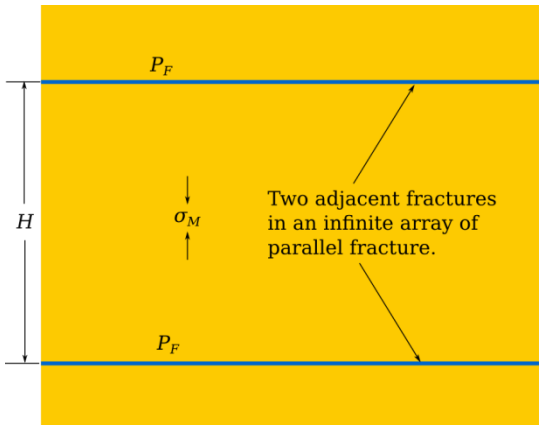


Figure 1: Two adjacent fractures in an infinite array of parallel fractures.

Note that σ_M is not a constant, but a function of σ_{Mi} and P_F. Assume under the initial condition (effective normal stress being σ_{Mi}) the mechanical aperture width is w_i; at arbitrary effective joint stress σ'_J the mechanical aperture width becomes w. We define the secant unloading joint stiffness to be

$$\bar{k}_n = \frac{\sigma_{MJ} - \sigma'_J}{w(\sigma'_J) - w_i} \quad (2)$$

The normal stiffness of a joint k_n is conventionally defined using the zero-normal stress state as the reference state, whereas we use the zero-fluid pressure state as the reference. The compression experienced by the rock body between two neighboring fractures due to a matrix stress increase from σ_{Mi} to σ_M should be the same as the joint dilation due to the corresponding effective stress change from σ_{Mi} to σ'_J, namely

$$w(\sigma'_J) - w_i = \frac{(\sigma_M - \sigma_{Mi})H}{E'} \quad (3)$$

where E' is the confined stiffness of the rock matrix as

$$E' = \frac{E(1-\nu)}{(1-2\nu)(1+\nu)} \quad (4)$$

with E and ν being the Young's modulus and Poisson's ratio of the rock, respectively. We introduce a length scale

$$\bar{h}_n = E'/\bar{k}_n \quad (5)$$

so that the closure of the joint between the reference stress state and the current effective stress state is the same as the compression of a layer of virtual rock mass of thickness $\bar{k}_n$ experiencing the same stress change.

By plugging equations (1) and (2) into equation (3), we can obtain the increment of rock matrix stress $\Delta\sigma_M = \sigma_M - \sigma_{Mi}$ as

$$\Delta\sigma_M = \begin{cases} P_F/(1+H/\bar{h}_n) & \text{if } P_F \le \sigma_{Mi}(1+\bar{h}_n/H) \\ P_F - \sigma_{Mi} & \text{otherwise} \end{cases} \quad (6)$$

which indicates that the fracture will be completely open if the fluid pressure is higher than a threshold value of $\sigma_{Mi}(1+\bar{h}_n/H)$. When the fracture is still partially closed, the rock matrix stress increment is only a small portion of the fluid pressure increment. A survey of rock mechanics literature (e.g. Bandis et al., 1983; Barton et al., 1985; Cook, 1992) found that $\bar{h}_n$ is generally within the range between tens to hundreds of millimeters for interlocked rocks compressed with a stress level typical of hydraulic stimulation applications. If the fracture spacing is in the range of a few meters to tens of meters, then the rock matrix stress increment is a relatively small percentage of the fluid pressure increment. Therefore, the stress shadowing effects for partially closed fractures can generally be ignored. We term this scenario "joint stiffness-dominated" regime for fracture flow.

On the other hand, however, as the fluid pressure exceeds the threshold value and the fractures are completely open, the opening of the fracture will not be governed by joint stiffness, but by the deformation

in the rock matrix instead. Under this condition, the stress shadowing effects mentioned in the first section dictates that a primary fracture will emerge and suppress the pressure propagation in neighbor fractures.

COUPLING JOINT MODEL WITH FLOW SOLVER

In order to investigate fluid pressure propagation in the joint stiffness-dominated regime in an arbitrary fracture network, we use the numerical model for hydraulic fracturing developed at the Lawrence Livermore National Laboratory (LLNL). This fully coupled hydro-geomechanical model has been described elsewhere (Fu et al., 2011) and will not be repeated here. However, the original model was formulated for the scenarios where the fractures are completely open. To simulate the cases with partially closed fractures, some modifications are necessary. The weak stress shadowing effect allows us to directly incorporate joint closure models into the finite volume flow solver.

Fractures are discretized into interconnected flow cells in the flow solver. The permeability of each flow cell is a function of the hydraulic aperture width and the fluid storage volume of a cell is related to the mechanical aperture size. It is well known that the hydraulic aperture width is highly correlated with the mechanical aperture size as effective stress evolves, with the former generally smaller than the latter (Cook, 1992). However, their difference is ignored in this preliminary study. In each time step of solving the network flow, the fluid mass into and out of each flow cell is calculated and subsequently, the fluid mass in each cell is updated. Fluid pressure in each cell is calculated using the following equation-of-state

$$P_F = \begin{cases} K_F\left(1-\dfrac{\rho_{ref}L_C w}{m_C}\right) & \text{if } m_C/L_C w \geq \rho_{ref} \\ P_{vap} & \text{if } m_C/L_C w < \rho_{ref} \end{cases} \qquad (7)$$

where K_f is the bulk modulus of the fluid; ρ_{ref} is the reference density of this fluid, namely the density at zero or the datum pressure; L_c is the length (area in 3D) of the fluid cell and w is the aperture width, so $L_c w$ is the fluid storage volume of the cell; m_c is the fluid mass in this cell; P_{vap} is the temperature-dependent vapor pressure of this fluid which can be considered to be zero for the purpose of hydraulic stimulation modeling as the pumping pressure is many orders of magnitude higher than the vapor pressure. In the original model, the aperture width is calculated based on the deformation of the rock mass through a finite element solver. In the current study, instead we adopt the well known closure model by Bandis et al. (1983), which relates the aperture width w and joint effective normal stress as

$$\sigma' = \frac{w_{max}-w}{a-b(w_{max}-w)} \qquad (8)$$

where w_{max} is the aperture width at the zero-effective stress state, which is essentially the maximum joint closure in the original joint model; a and b are two material-specific constant. We plug equations (7) and (8) into equation (1) and obtain

$$\sigma_M - K_F\left(1-\frac{\rho_{ref}L_C w}{m_C}\right) = \frac{w_{max}-w}{a-b(w_{max}-w)} \qquad (9)$$

which can be solved as

$$w = w_{max} - \frac{Aa+Bb+1-[(Aa+Bb+1)^2-4AaBb]^{0.5}}{2Ab} \qquad (10)$$

where $A=K_F\rho_{ref}L_C/m_C$ and $B=\sigma_M-K_F+Aw_{max}$ are two constants to simplify the expression of the equations. With this equation, we can directly calculate the aperture width at each time step from the updated fluid mass and then obtain the fluid pressure using equation (7). The finite element solid solver is not required for joint stiffness-dominated fluid flow.

FLOW IN A SINGLE FRACTURE

We investigate fluid flow in a single fracture in this section. Because the strong coupling between fluid pressure, aperture volume, and aperture permeability, closed-form solutions cannot be derived.

A 100 meter long straight fracture is considered and it is discretized into 1,000 flow cells with L_C=0.1 m. In the initial condition where no fluid exists in the fracture, the normal stress along the fracture σ_n=10 and we ignore the total normal stress change due to pressurization of the fracture. In this state, the aperture width w_i=0.01 mm whereas w_{max}=0.1 mm corresponding to the zero-effective stress state. The closing behavior of the fracture is anchored by these two states and we can back-calculate the two constants in the joint model as

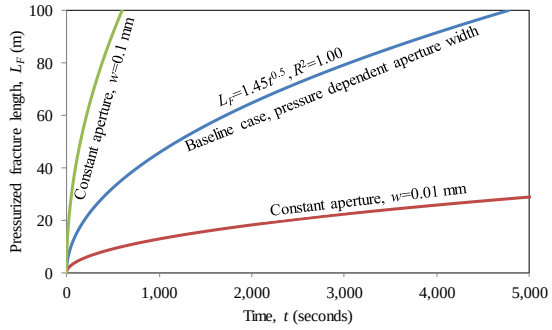
$$a = w_{max}\frac{w_{max}-w_i}{\sigma_n w_i} \text{ and } b = \frac{w_{max}-w_i}{\sigma_n w_i} \qquad (11)$$

The dynamic viscosity, bulk modulus (K_F), and reference density (ρ_{ref}) of the fluid are 0.001 Pa·s, 2.2 GPa, and 1,000 kg/m^3, respectively. All the above parameters remain the same for all the numerical examples in this paper unless stated otherwise.

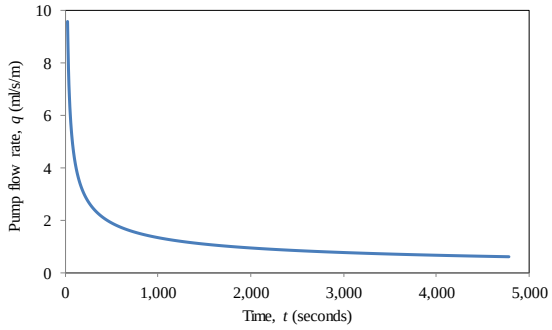
In the baseline scenario, we start pumping fluid with pressure P_{F0}=10 MPa into the fracture at time t=0. This is also the highest fluid pressure allowed by the joint stiffness-dominated regime. The length of the fracture that is pressurized by fluid L_F as a function of t is shown in Figure 2(a). A regression analysis finds that L_F is linearly proportion to the square root of t, with the regression equation and a perfect R^2 value shown in the figure. For an ideal case where the aperture width is a constant regardless of the fluid pressure, a closed-form solution exist between L_F and t as

$$L_F = w\left(\frac{P_0 t}{6\mu}\right)^{1/2} \tag{12}$$

which also indicates that L_F is a linear function of the square root of t. In Figure 2(a), two scenarios with pressure-independent aperture widths 0.1 mm and 0.01 mm are plotted. Since these two aperture widths are the upper and lower bounds of the aperture width in the baseline case, it is not surprising to see the propagation speed of the baseline case is somewhere between these two ideal cases.



(a)



(b)

Figure 2: Numerical simulation results for the baseline single fracture. (a).

The flow rate q at the pumping end of the fracture decreases as the fluid front propagates, as shown in Figure 2(b). This phenomenon has an important implication for the stimulation of a fracture network. It indicates that as the stimulation progresses, it will be more and more difficult to pump fluid into a single fracture. The flow tends to find alternative route, thereby stimulating other fractures in the network. On the other hand, if an open primary fracture has developed, the flow rate into this fracture increases as this fracture grows if the pump pressure remains constant. This single fracture will consume most of the fluid volume and make the stimulation of other fractures more difficult.

Two more simulations for the same single fracture but with lower pumping pressures, 5 MPa and 2 MPa, were performed and the results are shown in Figure 3. The effect of pumping pressure on the fluid front propagation rate is very significant. For all the three pumping pressures, L_F is always a linear function of the square root of time. We also implemented some other forms of the relationship between the effective normal stress and the aperture width in addition to the Bandis-Barton model, and found that this square root grow rate relationship is always valid.

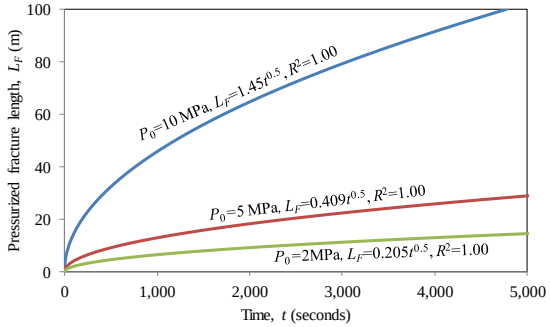


Figure 3: The effects of pumping pressure P_0 on the growth rate of L_F.

SELF-PROPPING THROUGH SHEAR DILATION

It is believed that a fracture network can be stimulated by the mechanism of shear dilation under the following conditions: 1) There exists significant shear stress long the fractures; and 2) the fluid pressure is high enough to induce shear slipping of the fractures as a result of the reduced effective stress. Predicting the amount of shear dilation is a challenging task, primarily due to the lack of experiment data that enable characterization of joint behaviors along the complex stress paths associated with hydraulic stimulation and the subsequent drawdown. The following simple phenomenological

empirical model is used in this study to represent the most important shear dilation behaviors associated with low pressure stimulation.

We introduce a variable, termed the stimulation factor S to quantify the extent to which a fracture has been stimulated through shear dilation. The aperture width is a function of not only the compressive effective stress σ', but also this stimulation factor S. If we assume the effects of σ' and those of S can be decoupled, S becomes a multiplier of the original joint model as

$$w = w(\sigma', S) = Sw(\sigma') \quad (13)$$

In the unstimulated state, $S=S_0=1$.We denote the three parameters in the joint model in this state as w_{max0}, a_0 and b_0, and the evolution of these parameters with S is as $w_{max} = w_{max0}S$ and $a = a_0S$ while b is a constant as $b=b_0$. We define the "excessive" shear stress along a fracture to be $\tau'=\tau_0-\sigma'\mu$, where τ_0 is the shear stress along the fracture in the initial state without hydraulic pressure and μ is coefficient of friction of the fracture. The stimulation factor S is assumed to be related to the greatest excessive shear stress τ'_{max} ever achieved by the fracture

$$S = \begin{cases} 1+\tau'_{max}(S_{max}-1)/\tau'_s & \text{if } \tau'\tau'_{max} < \tau'_s \\ S_{max} & \text{otherwise} \end{cases} \quad (14)$$

where S_{max} is the upper limit of S and S reaches S_{max} at excessive shear stress τ'_s. The above formulation dictates that an increase of the excessive shear stress can induce increase of S, but a decrease of τ' has no effect on S. In other words, the stimulation effects induced by the increase of fluid pressure will not be reversed when the pressure decreases after the stimulation. However, the aperture size is still a function of the effective stress as dictated by equation (8). The main effect of stimulation by shear dilation is to change the values of the constants in equation (8).

NUMERICAL EXAMPLE: STIMULATION OF A NATURAL FRACTURE NETWORK

In this section, we exercise the numerical model on a virtual reservoir. As shown in Figure 4, the simulation domain is 320 m wide (x from -160 m to 160 m) and 240 m tall (y from 0 to 240 m). There are two sets of joints (existing natural fractures) in this reservoir. The horizontal set has orientation angles with a uniform distribution between 10° and 30° where as the vertical set has orientation angles between 80° and 100°. Note that this 2D simulation domain should be considered as a plan view of the reservoir, so the term "vertical" refers to the direction within the image, not the vertical direction in a 3D space. All the fractures have lengths between 20 m and 60 m and the total length of fractures in the two sets are 8,300 m and 8,700 m respectively. The injection well is located near the middle point of the lower boundary of the domain as shown in Figure 4, so the simulation is on a half of the reservoir. The location of the production well is shown in Figure 4. The far field *in situ* stress applied is σ_{xx}=10 MPa, σ_{yy}=14 MPa, and σ_{xy}=0. Since the fractures usually do not exactly align with the coordinate system, shear stress dependent on the orientation angle exists along these fractures. Joint model parameters used are shown in Table 1 and parameters for the fluid phase are the same as the numerical examples for single fractures..

Table 1: Model parameters used in this study.

Parameter	Value
w_{max0}	0.2 mm
w_{i0}	0.02 mm
t'_s	3 MPa
S_{max}	3.0
μ	0.7

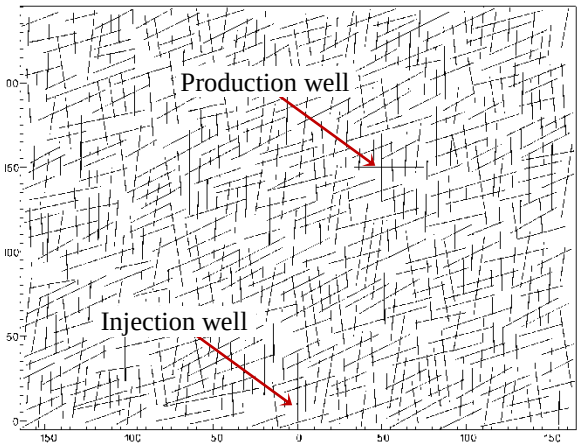


Figure 4: The effects of pumping pressure P_0 on the growth rate of L_F.

The injection pressure at the injection well is 10 MPa, the same as the minimal principal stress. The portions of the fracture network that is pressurized (with non-zero fluid fracture) at 20,000 seconds (5.6 hours) and 100,000 seconds (28 hours) after the injection has started are shown in Figure 5.

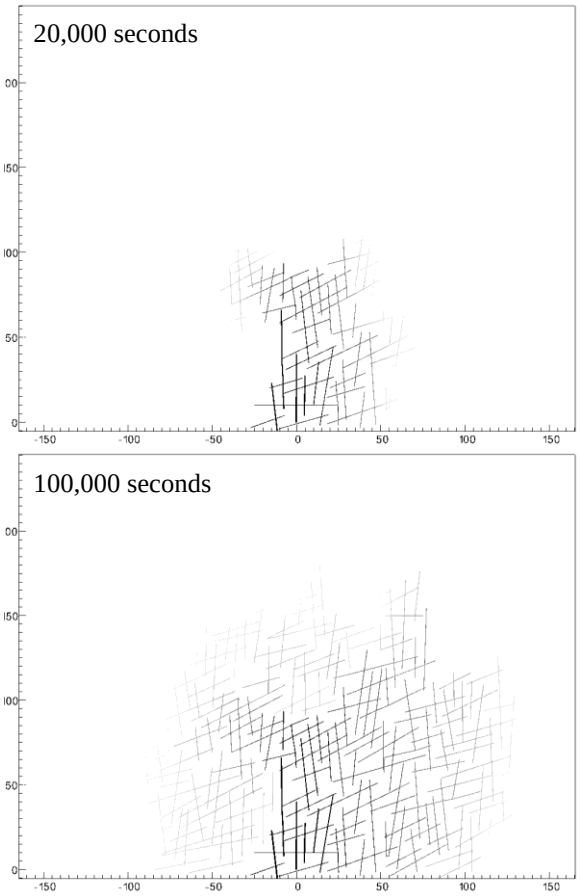


Figure 5: Pressurized fracture network 20,000 and 100,000 seconds after stimulation.

We simulate four scenarios (A to D) that share the same stimulation process in the first 100,000 seconds as described above but with different subsequent operations after 100,000 seconds. For cases A through C, we start pumping into the production well with 10 MPa fluid pressure from 100,000 seconds to 125,000 seconds. The objective is to stimulate the fractures near the production well. The difference between these three cases is the back pressure we use in the production state, being 0, 2 MPa, and 4 MPa for cases A, B, and C, respectively. A higher back pressure can increase the aperture width and permeability in the near-well region but on the other hand, it also decreases the pressure gradient from the injection well and the production well. In case D, we do not stimulate the region around the production well but directly apply 4 MPa of back pressure at 100,000 seconds.

The flow rates at the two wells for case A are shown in Figure 6. Negative flow rate means flow from the well into the reservoir and positive value means flow from the reservoir to the well. Because this is a 2D model, the flow rate is for unit-thickness reservoir. From the beginning of the stimulation (t=0) to 100,000 seconds, the absolute flow rate at the injection well continues to decreases, similar to what the single fracture model has shown. The fluid front reaches the production well at approximately 50,000 seconds, and fluid starts to flow out from that well, which is an artifact of the zero-pressure boundary condition given at the well. Fluid flow into the production well between t=100,000 and t=125,000 seconds during the stimulation through the production well. Once we lower the pressure to the back pressure (0 for case A), fluid starts to flow back into the well. The flow rate in the beginning is high due to the high pressure that has built up during the production well stimulation, and it soon reaches a relatively steady level when most of the fluid is supplied from the injection well. We terminate the simulation at 300,000 seconds because the fluid front has reaches the boundary of the simulation domain. At this moment, the injection rate is still slowly decreasing and the production rate is slowly increasing. The flow rate at t=300,000 seconds for all four scenarios are summarized in Table 2.

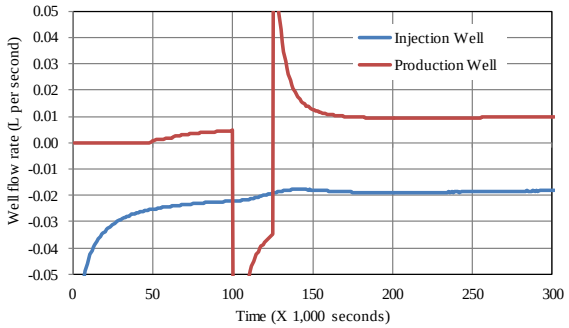


Figure 6: Flow rate at the two wells in scenario A.

Table 2: Absolute flow rate at t=300,000 seconds.

Scenario	Injection (L/s)	Production (L/s)
A	0.0183	0.00991
B	0.0197	0.00913
C	0.0197	0.00779
D	0.0193	0.00720

The fluid recovery ratios (production flow rate divided by injection rate) for the four scenarios are 54%, 46%, 40%, and 37%, respectively. The benefit of production well stimulation is apparent, but the back pressure seems to not only reduces flow rate but also decreases recovery rate. Placing more production wells should increase the recovery ratio, but this is to be studied in the future.

SUMMARY

In this study, we investigate the use of a numerical test bed to study the stimulation of existing fracture

networks with relatively low hydraulic pressure. We found that in this regime, the coupling between the flow and the solid phase can be considered local and the numerical model can be greatly simplified. The results show that low pressure stimulation can indeed stimulate the entire network, instead of propping a primary fracture as in high pressure stimulation. This paper only documents our initial effort along this path, and more realistic scenarios are to be studied.

AUSPICES AND ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Geothermal Technologies Program of the US Department of Energy for support of this work under the Enhanced Geothermal Systems Program. The authors also would like to acknowledge their collaborators at the Lawrence Livermore National Laboratory. Additional support provided by the LLNL LDRD project "Creating Optimal Fracture Networks" (#11-SI-006) is gratefully acknowledged. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This document has been released to unlimited external audience with an LLNL IM release number LLNL-CONF-523577.

REFERENCES

The Future of Geothermal Energy: Impact of Enhanced Geothermal Systems (EGS) on the United States in the 21st Century (2006). Research report by an MIT-led interdisciplinary panel.

Barton, N., Bandis, S., Bakhtar, K. (1985), "Strength, deformation and conductivity coupling of rock joints," *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, 22(3), 121-140.

Bandis, S., Lumsden, A., Barton, N. (1983), "Fundamentals of rock joint deformation," *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, 20(6), 249-268.

Cook, N.G.W. (1992), "Natural joints in rock: Mechanical, hydraulic and seismic behaviour and properties under normal stress," *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, 29(3),198-223..

Fu, P., Johnson, S.M., Hao, Y., and Carrigan, C.R. (2011), "Fully coupled geomechanics and discrete flow network modeling of hydraulic fracturing for geothermal applications." *The 36th Stanford Geothermal Workshop*, Jan. 31 – Feb. 2, 2011, Stanford, CA.

Johnson, S.M., and Morris, J.P. (2009), "Modeling hydraulic fracturing for carbon sequestration applications," *the 43rd US Rock Mechanics Symposium and the 4th US-Canada Rock Mechanics Symposium*, Asheville, NC, ARMA 09-30.

Willis-Richards, J., Watanabe, K., Takahashi, H. (1996), "Progress toward a stochastic rock mechanics model of engineered geothermal systems," *Journal of Geophysical Research*, 101(B8),17481–17.

A weighted Nitsche stabilized method for small-sliding contact on frictional surfaces

Chandrasekhar Annavarapu[a,*], Randolph R. Settgast[a], Scott M. Johnson[a], Pengcheng Fu[a], Eric B. Herbold[a]

[a] *Atmospheric, Earth, and Energy Division, Lawrence Livermore National Laboratory, Livermore, CA 94550, USA*

Abstract

We propose a weighted Nitsche framework for small-sliding frictional contact problems on three-dimensional interfaces. The proposed method inherits the advantages of both augmented Lagrange multiplier and penalty methods while also addressing their shortcomings. Algorithmic details of the traction update and consistent linearization in the presence of Nitsche terms are provided. Several benchmark numerical experiments are conducted and the results are compared with existing studies. The results are encouraging and indicate accurate satisfaction of the non-interpenetration constraint, stable tractions and asymptotic quadratic convergence of the Newton-Raphson method.

Keywords: frictional contact, frictional cracks, Nitsche, X-FEM, interface

1. Introduction

The mechanical response of frictional interfaces is of prime importance in many engineering applications ranging from crack-closure effects in micro-fractures to frictional sliding between rock joints and geological faults. These effects span length-scales that are orders of magnitude apart yet are equally significant at either end of this spectrum. Robust numerical strategies that

*Corresponding author: Chandrasekhar Annavarapu, Atmospheric, Earth, and Energy Division, Lawrence Livermore National Laboratory, 7000 East Avenue, L-286, Livermore, CA 94550, USA.

Email address: asc.sekhar@gmail.com, annavarapusr1@llnl.gov (Chandrasekhar Annavarapu)

enable modeling of these effects can have far-reaching consequences in guiding important engineering decisions such as predicting fatigue life [2], seismic hazard characterization and engineering fracture networks for efficient extraction of shale gas and geothermal energy [3, 4, 5, 6, 7, 8].

Fracture and interface problems in a Lagrangian framework typically require a constantly changing mesh topology to model discontinuous fields. This poses significant challenges for the finite element method. A variety of approaches have been proposed to address these issues including strong discontinuity approaches (see Simo *et al.* [9]) interface element techniques (see Xu and Needleman [10], Camacho and Ortiz [13], Settgast and Rashid [14]), discontinuous Galerkin methods (Radovitzky *et al.* [15]), and generalized and extended finite element methods (Duarte *et al.* [17], Moës *et al.* [16]). However, most of these methods suffer from numerical instabilities when confronted with the fundamental problem of crack closure (see Simone [48])that is hard to neglect for many physical applications. The key numerical issue in resolving crack closure effects concerns the enforcement of nonlinear contact constraints. The most commonly used approaches, *viz.* the penalty and variants of Lagrange multiplier methods for enforcing contact conditions either suffer from a lack of accuracy in the enforcement of the non-interpenetration condition or from spurious oscillations in contact stresses. For an exhaustive survey of the methods in computational contact mechanics we refer the interested reader to the monographs by Laursen [69] and Wriggers [70]. We limit the discussion to the contributions concerning elastostatic contact constraints on crack surfaces below.

For conventional interface elements, to enforce contact constraints, penalty based node-to-node or node-to-surface approaches have been developed by Gonçalves *et al.* [18], Day and Potts [19], Schellekens and De Borst [20], and Warner and Molinari [21] in both fully implicit and explicit time integration frameworks. Several studies also consider a phenomenological traction-separation relationship between the compressive tractions and interpenetrations (see Espinosa *et al.* [22, 23], An and Qin [24]) which essentially collapse into a penalty method for contact. However, all of these studies suffer from stress oscillations if the ratio of interface element stiffness with the stiffness of the element adjacent to it is not carefully chosen (see Day and Potts [19]). The LArge Time INcrement (LATIN) method of Ladeveze *et al.* [1] is another well-established numerical approach to model frictional contact problems (see Ladeveze *et al.* [25]).

Within an eXtended finite element method (X-FEM), the LATIN algo-

rithm has been applied to model crack face contact by Dolbow *et al.* [32]. Since then, the method has also been extended to three-dimensional problems by Gravouil *et al.* [65]. Several other Lagrange multiplier based strategies have also been developed. For instance, Kim *et al.* [34] proposed a Lagrange multiplier method where the interpolation space for the multipliers was constructed through an independent interface discretization. Giner *et al.* [38] used a segment-to-segment based contact approach with Lagrange multipliers to model crack face contact in 2D with bilinear quadrilateral elements. Nistor *et al.* [31] applied the method of Béchet *et al.* [56] and Hautefeuille *et al.* [55] to large sliding contact. Siavelis *et al.* [49] further extended this approach to model branched cracks. The challenge with Lagrange multiplier methods for this class of problems concerns the construction of an *inf-sup* stable Lagrange multiplier space [50, 51]. A naïve choice often results in spurious oscillations for the interfacial tractions.

A popular and commonly used strategy to circumvent the stability problems of the Lagrange multipliers is the penalty function method (see Perić and Owen [11]). In the context of crack surfaces, Liu and Borja [26], and Khoei and Nikbakht [57] developed penalty based approaches for frictional sliding under the small-sliding assumption. More recently, this approach has also been extended to problems with bulk plasticity (Khoei *et al.* [58], Liu and Borja [27]) and large sliding (Khoei and Mousavi [35], Liu and Borja [28]). Further, Liu and Borja [29] also proposed a projected polynomial pressure stabilized formulation for lower order elements, with the option of either Lagrange multipliers or penalty regularization for the contact constraints. More recently, Mueller-Hoppe *et al.* [39] proposed a penalty based contact formulation to prevent crack face penetration (with no tangential sliding) for hexahedral elements. The challenge with penalty-based formulations lies in identifying the "correct" parameters that yield well-conditioned equations while also satisfying the non-interpenetration conditions accurately. Interestingly, the numerical issues associated with traction oscillations are common to both the conventional interface elements where the mesh lines align with the interface and the enriched approaches where the interface is arbitrary with respect to the mesh [48].

We advocate the use of Nitsche's method [36] for this class of problems. In contact problems, Nitsche's method was proposed for frictionless contact problems by Wriggers and Zavarise [37]. For elastostatic contact, symmetric and non-symmetric variants of Nitsche's method have been proposed by Chouly *et al.* [40] and Renard [41]. For an analysis of the method estab-

lishing optimal convergence rates see Chouly and Hild [42] for frictionless problems and Chouly [43] for Tresca friction. Related stabilized methods include the Barbosa-Hughes stabilized method of Hild and Renard [44], and the variational multiscale approach of Masud *et al.* [45], Truster *et al.* [46]. Another closely related method based on a consistent perturbed Lagrangian formulation within the contact domain framework was developed by Oliver *et al.* [72] and Hartmann *et al.* [73] for large deformation frictional contact problems. Coon *et al.* [47] used Nitsche's technique to model earthquake rupture in an extended finite element framework. More recently, Annavarapu *et al.* [63, 64] proposed the use of a weighted form of Nitsche's approach to model two-dimensional frictional sliding problems over embedded interfaces. They found the weighted form to be advantageous for large contrasts in material properties and for high degree of anisotropy in meshes across the interface.

We extend the weighted Nitsche approach of Annavarapu *et al.* [63] to three-dimensional surfaces and trilinear hexahedral elements. There are many advantages to this approach over the aforementioned methods. First and foremost, it is a consistent primal method that circumvents the construction of an *inf-sup* stable multiplier space. Secondly, no additional degrees of freedom are introduced to the system-matrix and no augmentation loops are necessary to identify the correct multipliers. Finally, with the aid of numerical analysis the method does not contain any tunable parameters.

The rest of the paper is organized as follows. In Section 2, we describe the governing equations and the associated variational forms. In Section 3, we describe the theoretical framework for the proposed approach and the algorithmic treatment of contact tractions within the proposed framework. In Section 4, we discuss the discrete equations and the algorithmic tangent operators. In Section 5, we consider several benchmark examples to illustrate the performance of the method. Finally in Section 6, we offer concluding remarks and an outlook for the work.

2. Governing equations and Variational formulation

We begin by considering a domain Ω (an open subset in $\mathbb{R}^3$) and its boundary Γ as shown in Figure 1. Further, we consider Γ_c to represent a crack surface with Γ_c^1 and Γ_c^2 representing the initially coincident crack faces. The governing equations for small deformation elastostatics are now given in

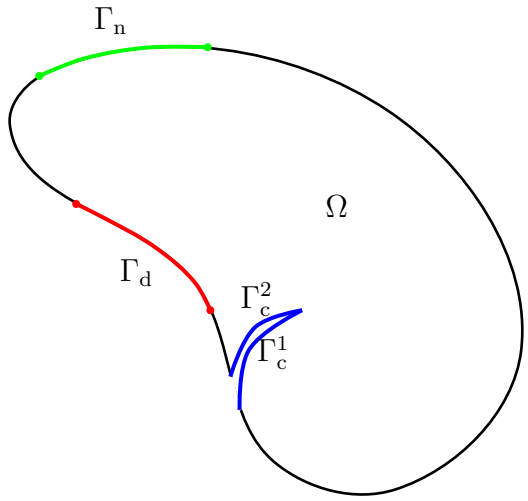


Figure 1: Notation for the model problem. Domain Ω, the Dirichlet boundary Γ_{d} and the Neumann boundary Γ_{n} are as shown. Γ_{c}^1 and Γ_{c}^2 represent the initially coincident crack faces. The complementary part of the boundary is traction free. The normal to the boundary of a domain is considered to point outwards from the domain.

indicial notation as:

$$\begin{aligned} \sigma_{ij,j} &= 0 \quad \text{in} \quad \Omega, \\ u_i &= u_i^{\mathrm{d}} \quad \text{on} \quad \Gamma_{\mathrm{d}}, \\ \sigma_{ij} n_j &= h_i \quad \text{on} \quad \Gamma_{\mathrm{n}}, \end{aligned} \tag{1}$$

where σ_{ij} and u_i denote the components of the stress and displacement fields in domain Ω, respectively, and n_j the components of the unit outward normal. The displacement is fixed to the prescribed field u_d on the Dirichlet portion of the boundary, and h_i denotes the prescribed traction on the Neumann portion of the boundary. We assume a linear elastic response for the constitutive relationship in the bulk domain:

$$\sigma_{ij} = C_{ijkl} u_{(k,l)} \quad \text{in } \Omega, \tag{2}$$

where C_{ijkl} denotes the fourth-order elasticity tensor, and $u_{(k,l)}$ is the symmetric gradient of the displacement field.

With respect to the constitutive relationship at the crack surface, we develop the proposed approach for crack surfaces under the small-sliding

assumption. First, we introduce a traction field by projecting the stress from each side of the crack surface so that:

$$t_i^{\mathrm{m}} = \sigma_{ij}^{\mathrm{m}} n_j^{\mathrm{m}} \quad \text{on } \Gamma_{\mathrm{c}}; \qquad \mathrm{m} = 1, 2. \tag{3}$$

We relate the tractions, t_i^2 and t_i^1 from both sides of the crack surface through a force-balance relation. Additionally, for convenience, we define a single traction field, t_i in the local coordinates $(\boldsymbol{n^2}, \boldsymbol{\tau^{2^1}}, \boldsymbol{\tau^{2^2}})$ of the crack face Γ_{c}^2 such that:

$$t_i = t_i^2 = -t_i^1 \quad \text{on } \Gamma_{\mathrm{c}}, \tag{4}$$

To define the interfacial kinematics and constitutive laws, we consider the following decompositions for the interfacial tractions and displacements in the local coordinates of the crack face Γ_{c}^2:

$$\begin{aligned} t_i &= t_N n_i + t_{\tau^1} \tau_i^1 + t_{\tau^2} \tau_i^2, \\ u_i^{\mathrm{m}} &= u_N^{\mathrm{m}} n_i + u_{\tau^1}^{\mathrm{m}} \tau_i^1 + u_{\tau^2}^{\mathrm{m}} \tau_i^2; \quad \mathrm{m} = 1, 2, \end{aligned} \tag{5}$$

For brevity, in the above, we dropped the superscripts associated with the crack face index and denote the local coordinates by $(\boldsymbol{n}, \boldsymbol{\tau^1}, \boldsymbol{\tau^2})$. We assume no gap in the normal component of the displacement field such that:

$$u_{\mathrm{N}}^1 = u_{\mathrm{N}}^2 \quad \text{on } \Gamma_{\mathrm{c}}. \tag{6}$$

However, we allow for tangential slip and the tangential component of the displacements are related to the tangential tractions through Coulomb's frictional behavior. We use $[\![u_j]\!] = u_j^2 - u_j^1$ to denote the jump in the displacement field (or gap) at the crack surface. The flow rule describing sliding and the yield (or slip) function for Coulomb's friction are written in rate form, and given by:

$$[\![\dot{\mathbf{u}}_\tau]\!] = \dot{\beta} \frac{\mathbf{t}_\tau}{||\mathbf{t}_\tau||}, \tag{7}$$

$$\phi(\mathbf{t}_\tau) = ||\mathbf{t}_\tau|| - \mu t_N, \tag{8}$$

where $\dot{\beta}$ represents the slip rate, $\phi(\mathbf{t}_\tau)$ represents the yield function and μ denotes the coefficient of friction between the crack faces. Finally, the Kuhn-Tucker complimentarity conditions and the consistency equation are given

by:

$$\dot{\beta} \geq 0, \qquad \phi(\mathbf{t}_\tau) \leq 0, \qquad \dot{\beta}\phi = 0,$$
$$\dot{\beta}\dot{\phi} = 0 \text{ (if } \phi = 0). \tag{9}$$

The first line in equation (9) specifies the requirements on the admissibility of the traction field. It also states that any slip only occurs on the yield surface. The second line represents a persistency condition which ensures that for the slip rate to be non-zero, the stress state must persist on the yield surface.

2.1. Variational form

The variational form of the governing equations described above can be derived as: Find $u_i \in \mathcal{U}_i$ such that:

$$\int_\Omega w_{(i,j)}\sigma_{ij} \, \mathrm{d}\Omega - \int_{\Gamma_\mathrm{c}^1} w_i^1 t_i^1 \, \mathrm{d}\Gamma - \int_{\Gamma_\mathrm{c}^2} w_i^2 t_i^2 \, \mathrm{d}\Gamma = \int_{\Gamma_\mathrm{n}} w_i h_i \, \mathrm{d}\Gamma \qquad \forall w_i \in \mathcal{V}_i, \tag{10}$$

From the traction continuity equation (4), we have: Find $u_i \in \mathcal{U}_i$ such that:

$$\int_\Omega w_{(i,j)}\sigma_{ij} \, \mathrm{d}\Omega - \int_{\Gamma_\mathrm{c}} [\![w_i]\!] t_i \, \mathrm{d}\Gamma = \int_{\Gamma_\mathrm{n}} w_i h_i \, \mathrm{d}\Gamma \qquad \forall w_i \in \mathcal{V}_i, \tag{11}$$

where $[\![w_i]\!]$ is the jump in the variations across the crack face and $\mathcal{U}_i$ and $\mathcal{V}_i$ are spaces of sufficiently smooth functions for the displacements and variations respectively. The first and third terms in equation (11) are standard. The second term represents the contact virtual work. In terms of the normal and tangential contributions, the contact integral can be written as:

$$\int_{\Gamma_\mathrm{c}} [\![w_i]\!] t_i \, \mathrm{d}\Gamma = \int_{\Gamma_\mathrm{c}} ([\![w_N]\!] t_N + [\![\mathbf{w}_\tau]\!] \cdot \mathbf{t}_\tau) \, \mathrm{d}\Gamma. \tag{12}$$

In the following section, we focus particular attention on this term and detail Nitsche's formulation for frictional contact.

3. Weighted Nitsche's method for frictional contact

3.1. Traction definition

The key idea of a Nitsche based formulation of contact is to evaluate the contact tractions in terms of the stress state of the bulk material. To that end, we define a weighted interfacial pressure and shear:

$$p_\gamma = n_i \langle \sigma_{ij} \rangle_\gamma n_j, \quad f_\gamma = \tau_i \langle \sigma_{ij} \rangle_\gamma n_j, \tag{13}$$

where $\langle \sigma_{ij} \rangle_\gamma = \gamma^1 \sigma^1_{ij} + \gamma^2 \sigma^2_{ij}$ represents a weighted average stress across the crack surface Γ_c. For a consistent formulation, the weights $\gamma^1 > 0$ and $\gamma^2 > 0$ are only required to satisfy $\gamma^1 + \gamma^2 = 1$. In the weighted Nitsche's method for sliding contact, the normal traction in (12) is evaluated as:

$$t_N = p_\gamma - \alpha_N [\![u_N]\!], \tag{14}$$

where $\alpha_N > 0$ is a stabilization parameter in the normal direction.

For algorithmic treatment of stick-slip behavior, it is convenient to additively decompose the tangential slip into a recoverable "elastic" and a non-recoverable "plastic" part so that:

$$[\![\mathbf{u}_\tau]\!] = [\![\mathbf{u}^{\mathbf{el}}_\tau]\!] + [\![\mathbf{u}^{\mathbf{pl}}_\tau]\!]. \tag{15}$$

A regularized version of Coulomb's frictional behavior is now stated as follows:

$$\phi(\mathbf{t}_\tau) = ||\mathbf{t}_\tau|| - \mu t_N \leq 0, \tag{16}$$

$$[\![\dot{\mathbf{u}}^{\mathrm{pl}}_\tau]\!] = \dot{\beta} \frac{\mathbf{t}_\tau}{||\mathbf{t}_\tau||}, \tag{17}$$

$$\dot{\mathbf{t}_\tau} = \dot{f}_\gamma \boldsymbol{\tau} - \alpha_\tau ([\![\dot{\mathbf{u}_\tau}]\!] - [\![\dot{\mathbf{u}}^{\mathrm{pl}}_\tau]\!]), \tag{18}$$

$$\dot{\beta} \geq 0, \qquad \dot{\beta}\phi = 0, \qquad \dot{\beta}\dot{\phi} = 0. \tag{19}$$

where the tangential plane $\boldsymbol{\tau}$ is spanned by unit orthonormal vectors $(\boldsymbol{\tau^1}, \boldsymbol{\tau^2})$ and $\alpha_\tau > 0$ is a stabilization parameter in the tangential direction. Although, a different parameter can be prescribed for each of the directions, here we choose an identical value for simplified representation. More discussion on

the particular choice for the stabilization parameters α_N, α_τ and the weights γ is conducted in Section 4.

Notice the similarity between the traction expressions (14) and (18) to augmented Lagrangian treatments of contact [33], where for frictional sliding the tractions are defined as

$$t_{\mathrm{N}} = \lambda_N - \epsilon_N [\![u_N]\!], \tag{20}$$

$$\dot{\mathbf{t}_\tau} = \dot{\boldsymbol{\lambda}}_\tau - \epsilon_\tau ([\![\dot{\mathbf{u}}_\tau]\!] - [\![\dot{\mathbf{u}}_\tau^{\mathrm{pl}}]\!]), \tag{21}$$

with λ_N, $\boldsymbol{\lambda_\tau}$ representing the Lagrange multipliers in the normal and tangential directions and ϵ_N, ϵ_τ representing the penalty parameters. The augmented Lagrange multiplier treatments are advantageous over pure penalty methods due to the consistency they lend the formulation. In theory, any non-zero value of ϵ_N and ϵ_τ results in the satisfaction of contact constraints. In practice, finite values are specified to avoid stability issues resulting from indefinite matrices. In the weighted Nitsche treatment, the multipliers are eliminated through their physical interpretation and consequently the approach retains the consistency properties of the Lagrange multiplier method. For a more theoretically rigorous discussion on the consistency of the formulation, we refer the interested reader to the paper by Chouly [43].

The primary difference between the weighted Nitsche's treatment and augmented Lagrangian methods is that the former does not introduce an additional field. No additional iterations are required to identify the correct multipliers as the pressure p_γ and shear f_γ are defined directly in terms of the bulk stress-state. Finally, as a consequence of the consistency of the method, the parameters α_N and α_τ do not serve so much to enforce the non-interpenetrability constraint as they do to ensure numerical stabilization for the method. Another consequence of consistency is that unlike penalty regularizations, the "elastic" component of slip is minimized and consequently we get a closer approximation to Coulomb's behavior than a regularized law.

While we notice that the proposed variational formulation is non-symmetric, we recall that for frictional contact problems a symmetric bilinear form is not necessarily advantageous. The coupling in normal and tangential directions for Coulomb's friction results in a naturally non-symmetric tangent stiffness even for symmetric bilinear forms (see [69, 70]).

3.2. Algorithmic treatment of frictional contact conditions

The external load is applied incrementally and the contact integral (12) is evaluated at load step n+1 as:

$$\int_{\Gamma_c} [\![w_i]\!] t_i \, \mathrm{d}\Gamma = \int_{\Gamma_c} ([\![w_N]\!] t_N^{\mathrm{n+1}} + [\![\mathbf{w}_\tau]\!] \cdot \mathbf{t}_\tau^{\mathrm{n+1}}) \, \mathrm{d}\Gamma. \tag{22}$$

For frictional sliding without an opening mode, the normal tractions at load step n+1 are given by:

$$t_{\mathrm{N}}^{\mathrm{n+1}} = p_\gamma^{\mathrm{n+1}} - \alpha_N [\![u_N^{\mathrm{n+1}}]\!], \tag{23}$$

The equations (16)-(19) are integrated using a backward Euler integration scheme and the tractions are updated using the return mapping strategy. The basic algorithmic framework is described below (see Simo and Hughes [68] for a detailed description of the return mapping approach). To begin with, a trial stick state is calculated as:

$$\begin{aligned} \mathbf{t}_\tau^{\text{trial, n+1}} &= f_\gamma^{\mathrm{n+1}} \boldsymbol{\tau} + \alpha_\tau ([\![\mathbf{u}_\tau]\!]^{\mathrm{n+1}} - [\![\mathbf{u}_\tau^{\mathbf{pl}}]\!]^{\mathrm{n}}), \\ \phi^{\text{trial, n+1}} &= ||\mathbf{t}_\tau^{\text{trial, n+1}}|| - \mu t_N^{\text{trial, n+1}}, \end{aligned} \tag{24}$$

Finally, the tractions are projected on to the yield surface through a return map:

$$\mathbf{t}_\tau^{\mathrm{n+1}} = \mathbf{t}_\tau^{\text{trial, n+1}} - \alpha_\tau \Delta\beta \frac{\mathbf{t}_\tau{}^{\text{trial, n+1}}}{||\mathbf{t}_\tau{}^{\text{trial, n+1}}||}. \tag{25}$$

The magnitude of slip is given as:

$$\Delta\beta = \begin{cases} 0 & \text{if } \phi^{\text{trial, n+1}} \leq 0, \\ \dfrac{\phi^{\text{trial, n+1}}}{\alpha_\tau} & \text{otherwise.} \end{cases} \tag{26}$$

Substituting equations (23) and (25) into (22) completes the weighted Nitsche formulation for frictional contact.

4. Discretization and Algorithmic Tangent Operators

The domain Ω is discretized into a set of non-overlapping regions Ω_e. We introduce the spaces $\mathcal{U}_i^h \subset \mathcal{U}_i$ and $\mathcal{V}_i^h \subset \mathcal{V}_i$ as finite-dimensional approximations to the solution and weighting spaces. We follow the standard

Bubnov-Galerkin approximation and assume $\mathcal{U}^h$ and $\mathcal{V}^h$ to be identical with Dirichlet conditions built into the the solution space $\mathcal{U}^h$. The displacement interpolation can now be constructed as:

$$\mathbf{u}^{\mathbf{h}} = \sum_I \mathbf{N}_I \mathbf{u}_I \tag{27}$$

where $\mathbf{N}_I$ are the nodal shape functions constructed from piecewise continuous polynomial functions and $\mathbf{u}_I$ are the nodal degrees of freedom.

Introducing the above approximation (and an identical approximation for the weighting functions) into the variational form (11), it is easy to obtain the following discrete statement of equilibrium in the residual form:

$$\mathbf{R}(\mathbf{u}) = \mathbf{F}_{\text{ext}}(\mathbf{u}) - \mathbf{F}_{\text{int}}(\mathbf{u}) = 0, \tag{28}$$

where the external force vector is given as:

$$\mathbf{F}_{\text{ext}}(\mathbf{u}) = \mathop{\mathbf{A}}_{e} \int_{\Gamma_{\text{n}e}} \mathbf{N}^{\text{T}} \mathbf{h}^{\text{m}} \, \text{d}\Gamma_e, \tag{29}$$

where $\mathbf{A}$ denotes the assembly operator. The internal force vector has bulk and contact contributions so that:

$$\mathbf{F}_{\text{int}}(\mathbf{u}) = \mathbf{F}^{\text{b}}_{\text{int}}(\mathbf{u}) + \mathbf{F}^{\text{c}}_{\text{int}}(\mathbf{u}). \tag{30}$$

The bulk contribution can be written as follows:

$$\mathbf{F}^{\text{b}}_{\text{int}}(\mathbf{u}) = \mathop{\mathbf{A}}_{e} \int_{\Omega_e} (\mathbf{B}^{\text{T}} \mathbf{D} \mathbf{B}) \mathbf{u}_{\text{e}} \, \text{d}\Omega_e, \tag{31}$$

where the matrix $\mathbf{B}$ contains the shape function derivatives, the matrix $\mathbf{D}$ represents the elasticity tensor in Voigt notation, and $\mathbf{u}_{\text{e}}$ is the local vector of nodal unknowns. Finally, the contact contribution to the internal force vector is obtained:

$$\mathbf{F}^{\text{c}}_{\text{int}} = \mathop{\mathbf{A}}_{e} \int_{\Gamma_{ce}} \mathbf{N}^{\text{T}} \; \mathbf{t}(\llbracket \mathbf{u} \rrbracket) \, \text{d}\Gamma_e, \tag{32}$$

where $\mathbf{t}(\llbracket \mathbf{u} \rrbracket)$ is the contact traction, updated as described in Section 3.2. We solve the nonlinear set of equations (28) at each load step using the Newton-Raphson iterative scheme. We linearize the internal force vector,

$\mathbf{F}^{\mathrm{b},\,\mathrm{n+1}}_{\mathrm{int},\,(\mathrm{k})} + \mathbf{F}^{\mathrm{c},\,\mathrm{n+1}}_{\mathrm{int},\,(\mathrm{k})}$, about the current state, defined by $\mathbf{u}^{\mathrm{n+1}}_{(\mathrm{k})}$, using a first-order Taylor series expansion to obtain:

$$\mathbf{K}^{\mathrm{n+1}}_{(\mathrm{k})} \Delta \mathbf{u}^{\mathrm{n+1}}_{(\mathrm{k+1})} = \mathbf{F}^{\mathrm{n+1}}_{\mathrm{ext}} - (\mathbf{F}^{\mathrm{b},\,\mathrm{n+1}}_{\mathrm{int},\,(\mathrm{k})} + \mathbf{F}^{\mathrm{c},\,\mathrm{n+1}}_{\mathrm{int},\,(\mathrm{k})}). \tag{33}$$

We solve for the incremental nodal displacement, $\Delta \mathbf{u}^{\mathrm{n+1}}_{(\mathrm{k+1})}$, at the k-th iteration. For brevity, subsequently, we omit the superscript, k, and the subscript, n+1, denoting the iteration and load counters respectively. The tangent matrix, at the (n+1)-th load step and the k-th iteration is now denoted by $\mathbf{K}$, such that:

$$\mathbf{K} = \frac{\partial (\mathbf{F}^{\mathrm{b}}_{\mathrm{int}} + \mathbf{F}^{\mathrm{c}}_{\mathrm{int}})}{\partial \mathbf{u}}. \tag{34}$$

The linearization of bulk contribution yields:

$$\frac{\partial \mathbf{F}^{\mathrm{b}}_{\mathrm{int}}}{\partial \mathbf{u}} = \mathop{\mathsf{A}}_{e} \int_{\Omega_e} \mathbf{B}^{\mathrm{T}} \mathbf{D} \mathbf{B} \, \mathrm{d}\Omega_e. \tag{35}$$

The linearization of the contact contribution yields:

$$\frac{\partial \mathbf{F}^{\mathrm{c}}_{\mathrm{int}}}{\partial \mathbf{u}} = \mathop{\mathsf{A}}_{e} \int_{\Gamma_{c_e}} \mathbf{N}^{\mathrm{T}} \, \frac{\partial \mathbf{t}(\llbracket \mathbf{u} \rrbracket)}{\partial \mathbf{u}} \, \mathrm{d}\Gamma_e, \tag{36}$$

where the contact traction $\mathbf{t}(\llbracket \mathbf{u} \rrbracket)$ depends on the traction update (25). It is easy to obtain the following block structure for the linearized contact tangent matrix:

$$\mathbf{K} \;=\; \begin{bmatrix} \mathbf{K}^{\mathrm{c},1}_{\mathrm{d}} & \mathbf{K}^{\mathrm{c},1}_{\mathrm{od}} \\ \\ \mathbf{K}^{\mathrm{c},2}_{\mathrm{od}} & \mathbf{K}^{\mathrm{c},2}_{\mathrm{d}} \end{bmatrix}, \tag{37}$$

where 1 and 2 represent the local indices of the contact face pairs. While still under a stick state, from (26) and (25), $\mathbf{t}(\llbracket \mathbf{u} \rrbracket) = \mathbf{t}^{\mathrm{trial}}(\llbracket \mathbf{u} \rrbracket)$. Recognizing from (25) that the trial traction additively decomposes into the Nitsche consistency and stabilization parts, evaluating, $\partial \mathbf{t}^{\mathrm{trial}}(\llbracket \mathbf{u} \rrbracket)/\partial \mathbf{u}$ results in the following expressions for the local matrices:

$$\begin{aligned} \mathbf{K}^{\mathrm{c,m}}_{\mathrm{d}} &= \mathop{\mathsf{A}}_{e} \mathbf{k}^{\mathrm{stab}}_{\mathbf{e}} - \mathop{\mathsf{A}}_{e} \gamma^{\mathrm{m}}_{e} \mathbf{k}^{\mathrm{nit},\,\mathrm{m}}_{\mathbf{e}} \quad \text{for m} = 1, 2, \\ \mathbf{K}^{\mathrm{c},1}_{\mathrm{od}} &= -\mathop{\mathsf{A}}_{e} \mathbf{k}^{\mathrm{stab}}_{\mathbf{e}} + \mathop{\mathsf{A}}_{e} \gamma^{2}_{e} \mathbf{k}^{\mathrm{nit},\,2}_{\mathbf{e}}, \\ \mathbf{K}^{\mathrm{c},2}_{\mathrm{od}} &= -\mathop{\mathsf{A}}_{e} \mathbf{k}^{\mathrm{stab}}_{\mathbf{e}} + \mathop{\mathsf{A}}_{e} \gamma^{1}_{e} \mathbf{k}^{\mathrm{nit},\,1}_{\mathbf{e}}. \end{aligned} \tag{38}$$

The entries of the local matrices are given as follows:

$$\begin{aligned} k_{ij}^{\text{stab}} &= \int_{\Gamma_{ce}} N_j \frac{\partial t_i^{\text{stab, trial}}}{\partial u_j} \, \mathrm{d}\Gamma_e; \quad \frac{\partial t_i^{\text{stab, trial}}}{\partial u_j} = \alpha_{ij} N_j, \\ k_{ij}^{\text{nit, m}} &= \int_{\Gamma_{ce}} N_j \frac{\partial t_i^{\text{nit, m, trial}}}{\partial u_j} \, \mathrm{d}\Gamma_e; \quad \frac{\partial t_i^{\text{nit, m, trial}}}{\partial u_j} = n_l^{\text{m}} D_{lk}^{\text{m}} B_{kj}^{\text{m}} N_j. \end{aligned} \tag{39}$$

When in perfect contact, the stabilization parameters in the normal and tangential directions are chosen identically with no coupling between the normal and tangential directions such that $\alpha_{ij} = \alpha\delta_{ij}$. The stabilization parameter α and the interfacial weights γ_e^{m} play a key role in the numerical performance of the method [61, 62]. These are evaluated locally as detailed in Annavarapu *et al.* [61]. The central idea proposed there was to identify the weights that result in the smallest possible value of the stabilization parameter while ensuring the coercivity of discrete forms. For bilinear and trilinear elements, while a rigorous coercivity criteria requires a local eigenvalue computation (see Embar *et al.* [52], Harari and Shavelzon [53]), we utilize the algebraic estimates available for constant strain elements for these elements as well to save computational expense. In practice, the algebraic estimate obtained for constant strain elements might slightly underestimate the stabilization necessary for bilinear quadrilaterals and trilinear hexahedral elements (Sanders *et al.* [54]). However, we did not notice instability in the numerical examples we investigated.

When slipping, the contributions from the slip vector in the tangential plane also need to be accounted for during linearization so that (in indicial notation):

$$\frac{\partial t_{\tau_i}}{\partial u_j} = \frac{\partial t_{\tau_i}^{\text{trial}}}{\partial u_j} - \frac{\partial}{\partial u_j}\left(\Delta\beta\alpha m_i\right); \quad m_i = \frac{t_{\tau_i}^{\text{trial}}}{||\mathbf{t}_\tau^{\text{trial}}||}; \quad \text{for } i = 1, 2; j = n, \tau^1, \tau^2. \tag{40}$$

Substituting the expressions for $\Delta\beta$ from (26) and using the chain rule of differentiation, we get the expression for the algorithmic tangent operator:

$$\frac{\partial t_{\tau_i}}{\partial u_j} = \frac{\mu t_N^{\text{trial}}}{||\mathbf{t}_\tau^{\text{trial}}||}\left(\frac{\partial t_{\tau_i}^{\text{trial}}}{\partial u_j} - m_i m_k \frac{\partial t_{\tau_k}^{\text{trial}}}{\partial u_j}\right) + \mu m_i \frac{\partial t_N^{\text{trial}}}{\partial u_j}. \tag{41}$$

Note that, on transforming the trial stiffness given in (39) to the local coordinates $(\boldsymbol{n^1}, \boldsymbol{\tau^1}, \boldsymbol{\tau^2})$, we already know all the terms that appear on the right

hand side of (41) and the tangent operator can now be easily evaluated. The additive decomposition property of the trial traction can again be utilized to update the Nitsche consistency and stabilization stiffnesses separately. Also, note that in the normal direction, no update is necessary as the trial and true tractions are the same.

5. Numerical examples

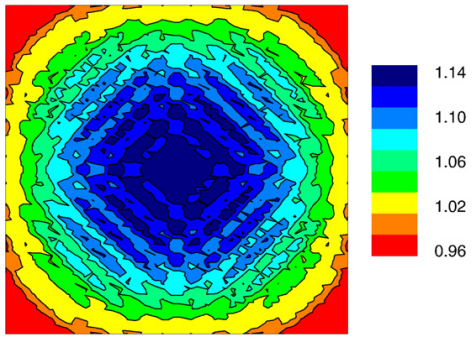
In this section, we consider several benchmark numerical examples to validate the performance of the proposed approach. Unless otherwise specified, we use a weighted form of Nitsche's method in all our simulations. We provide plots for bulk displacements and contact tractions and compare these results with existing studies where applicable. The bulk material follows linear elasticity while the contact interface is assumed to follow Coulomb's frictional behavior. The nonlinear equilibrium equations are solved iteratively using the Newton-Raphson method. Convergence is measured in the energy norm and a tolerance of 10^{-16} is specified for all the examples considered. Finally, the numerical simulations are conducted using GEOS [12], a massively-parallel multiphysics simulation software developed at Lawrence Livermore National Laboratory.

5.1. Frictionless contact between two elastic blocks

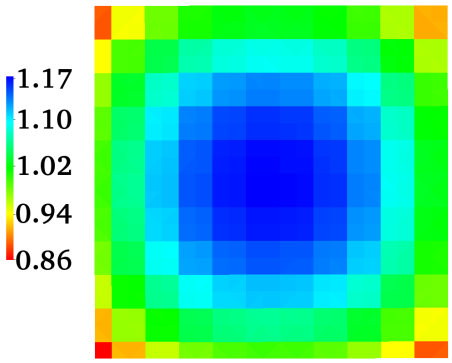
As a first example, we consider a planar crack surface under frictionless sliding investigated earlier by Liu and Borja [29]. We consider an elastic cube of unit length with a crack surface located at z = 0.5 m and extending through the length of the cube in the x and y dimensions. Unlike the earlier study by Liu and Borja [29], here, we assume that the mesh lines align with the crack surface. The bulk material obeys linear elasticity and has a Young's modulus of $E = 10$ GPa and a Poisson's ratio of $\nu = 0.3$.

The boundary conditions are such that the surface z=0 m is constrained in all directions while the top surface z=1 m is constrained laterally and a uniform displacement of $u_z = -0.1$ m is applied to load the cube in compression. We conduct the simulations using both constant strain tetrahedral and trilinear hexahedral elements with 21 divisions in x, y and z directions in each case. As expected for frictionless sliding, the method converged in three Newton iterations for both tetrahedral and hexahedral elements.

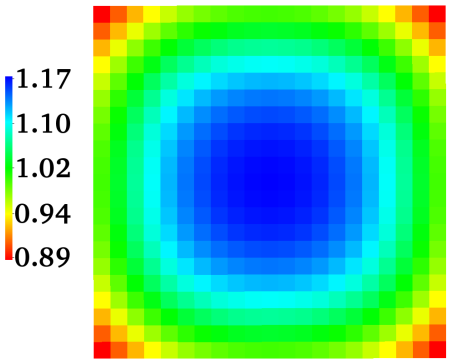
In Figure 2, we compare the normal pressures at the crack surface obtained using Nitsche's method and unstabilized penalty and Lagrange mul-



(a) Normal pressures (in GPa) at the crack surface for the unstabilized Lagrange multiplier and penalty methods from Liu and Borja [29]



(b) Normal pressures (in GPa) at the crack surface with Nitsche's method and constant strain tetrahedrons



(c) Normal pressures (in GPa) at the crack surface with Nitsche's method and trilinear hexahedral elements

Figure 2: Comparison of normal contact pressures at the crack surface obtained using Nitsche's method (bottom) and unstabilized penalty and Lagrange multiplier methods from Liu and Borja [29] (top)

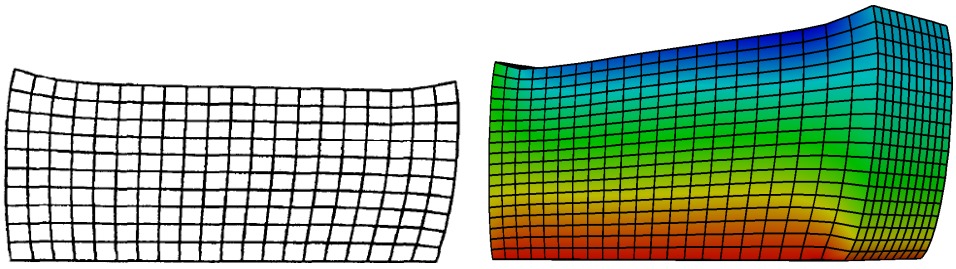
tiplier approaches from the earlier study by Liu and Borja [29]. Figure 2(b) shows the normal pressures obtained using linear tetrahedral elements while in Figure 2(c) we plot the pressures obtained using trilinear hexahedral elements. Clearly, the pressures obtained using Nitsche's method are smooth and do not exhibit the instability observed in unstabilized penalty and Lagrange multiplier approximations.

5.2. Sliding of an elastic block on a rigid surface

As a next example, we consider an elastic block sliding on a rigid foundation considered previously in Oden and Pires [60], Wriggers *et al.* [59], Simo and Laursen [33], Armero and Petocz [30] and Annavarapu *et al.* [63]. This benchmark is standard for frictional contact problems and serves as an important validation study. For completeness, we describe the problem setup and boundary conditions below.

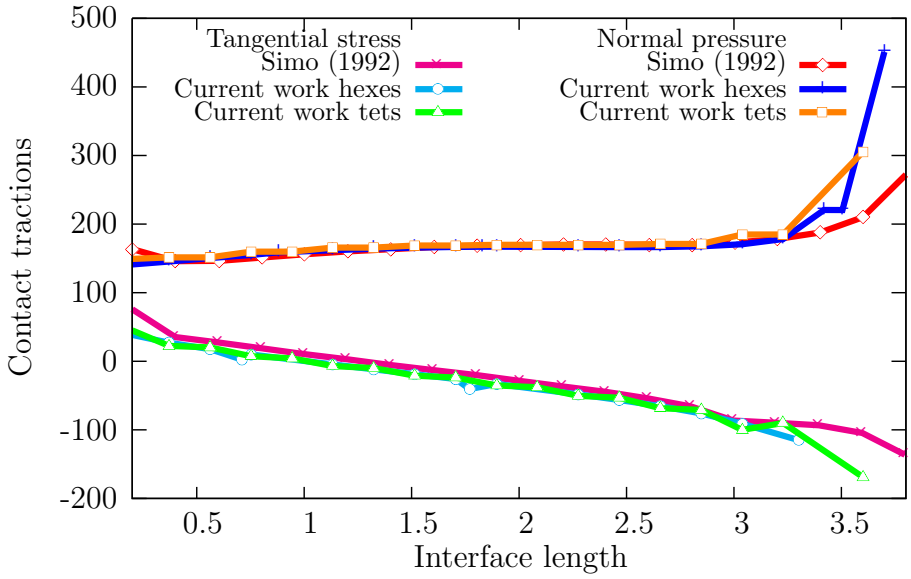
The computational domain is a rectangular parallelepiped spanning $(0, 4)\times(-0.4, 2)\times(0, 1)$ with an interface at $y = -0.057$ and an outward normal in the $(0, -1, 0)$ direction separating the elastic and rigid blocks. The elastic block is considered to have a Young's modulus of $E = 1000$ units and a Poisson's ratio of $\nu = 0.3$ while the rigid block has a Young's modulus of $E = 10^{12}$ units and $\nu = 0.0$. Coulomb's frictional behavior is assumed for frictional sliding between the blocks with a frictional coefficient of $\mu = 0.5$. The loading is prescribed such that the elastic block is pressed down in the negative y-direction on the surface $y = 2$ and pulled in the positive x-direction on the surface $x = 4$. The rigid block is constrained on the surface $y = -0.4$ in both x and y directions. Further, to reproduce the plane-strain conditions of previous studies, we constrain the surfaces $z = 0$ and $z = 1$ in the z-direction. For our studies, we consider two separate discretizations employing 15876 constant strain tetrahedral and 4410 trilinear hexahedral elements. The method converges in four Newton iterations and the convergence profile of Newton-Raphson iterative scheme is shown in Table 1. Also, as reported for the two-dimensional studies before in [63], classical Nitsche's method fails to converge for this problem. A more detailed comparison between classical and weighted Nitsche approaches is conducted in [61].

In Figures 3(a) and 3(b), we compare the deformed geometry obtained from Simo and Laursen [33] with the current study. Further, in Figure 3(c), we plot the contact tractions obtained from the current study using both tetrahedral and hexahedral elements and compare them with those presented by Simo and Laursen [33]. As we can see from the results, both the deformed



(a) Deformed geometry reproduced from Simo and Laursen [33].

(b) Deformed geometry using Nitsche's method. The plotted displacement contours are the y-components of the displacement.



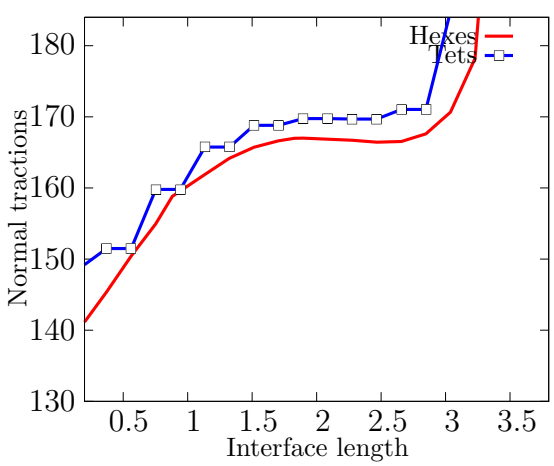
(c) Contact tractions using Nitsche's method.

Figure 3: Deformed geometry and contact tractions for the elastic block sliding on a rigid surface.

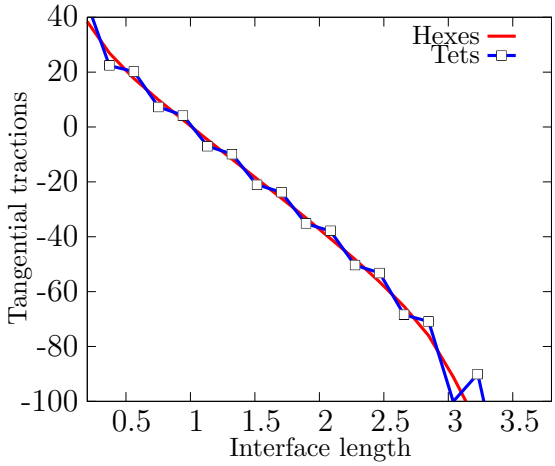
Table 1: Newton-Raphson convergence behaviour of Nitsche's method for the sliding block problem

Iteration number	Energy norm	
	Weighted Nitsche's method	Classical Nitsche's method
1	1.00e+00	1.00e+00
2	9.33e-07	5.40e-08
3	1.99e-14	1.07e-05
4	8.45e-30	6.41e-04
5	–	5.65e-03
6	–	1.26e-01
7	–	1.43e+00
8	–	1.66e+01
9	–	1.57e+03
10	–	2.40e+04
11	–	DNC

* DNC: Did not converge in 50 iterations.



(a) Comparison of contact pressures for Nitsche's method obtained using tetrahedral and hexahedral elements.



(b) Comparison of tangential stresses for Nitsche's method obtained using tetrahedral and hexahedral elements.

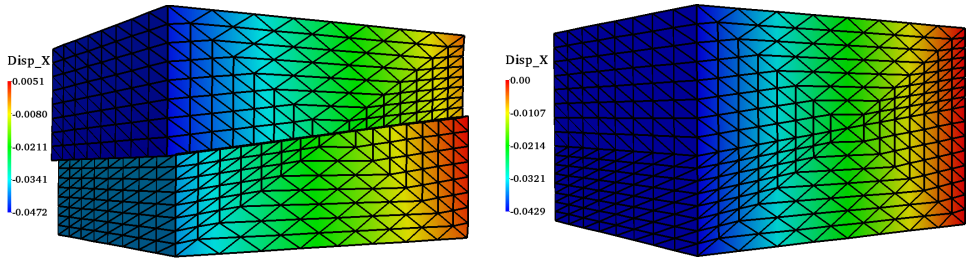
Figure 4: Comparison of contact tractions for Nitsche's method obtained using tetrahedral and hexahedral elements.

geometry and the contact tractions obtained from the current study are in excellent agreement with existing studies. In order to further investigate the minor oscillations in contact tractions for Nitsche's method with constant strain triangular elements earlier reported in Annavarapu *et al.* [63], we plot the normal and tangential tractions obtained using both tetrahedral and hexahedral elements and compare them in Figure 4. Clearly, we can see that the hexahedral elements result in a smoother traction profile and the tetrahedral elements continue to exhibit the minor oscillatory pattern reported earlier. We can thus conclude that the oscillatory pattern results from the poor stress approximation yielded by tetrahedral elements rather than any inherent instability in Nitsche's method.

5.3. Compressive loading of a plate with an inclined interface

We next investigate the stick-slip response of a plate loaded in compression with an interface inclined at an angle of θ to the x-y plane and extending throughout the z-dimension. This problem has been investigated earlier in plane-strain conditions by Dolbow *et al.* [32], Kim *et al.* [34] and Annavarapu *et al.* [63] and provides an easy way to test the method's capability in modeling Coulomb's friction. The problem is analogous to that of a rigid block resting on an inclined surface and predicts stick response when the coefficient of friction $\mu > \tan\theta$ and a slip response when $\mu < \tan\theta$. The problem setup is identical to the one described in previous studies. Appropriate boundary conditions are prescribed to eliminate rigid body modes and $y = y_{max}$ surface is loaded in compression by prescribing a displacement of $u_y = -0.1$. Further, to reproduce plane-strain conditions, we also constrain the surfaces $z = z_{min}$ and $z = z_{max}$ in the z-directions. The domain is meshed with 4627 constant strain tetrahedral elements using the open source mesh generation software Gmsh [71] such that the mesh lines align with the interface. The orientation of the interface is specified such that $\tan\theta = 0.2$. Further, the bulk material has a Young's modulus of $E = 1000$ units and a Poisson's ratio of $\nu = 0.3$. We run two separate computations by changing the coefficient of friction μ. In the first case, the coefficient of friction is chosen as $\mu = 0.19$ such that slipping response is predicted at the interface and in the second case we choose $\mu = 0.21$ so that we expect a stick state.

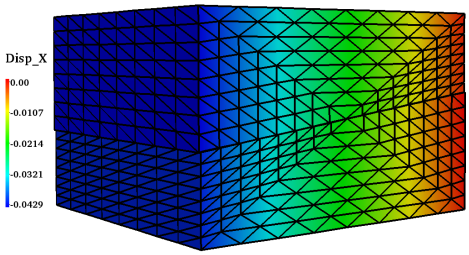
The method converged in four iterations when slipping and, as expected, in two iterations while sticking. The results of these simulations are plotted in Figure 5. The discontinuity in the x-displacement contours is evident in Figure 5(a) when the coefficient of friction $\mu = 0.19 < \tan\theta$ while we also



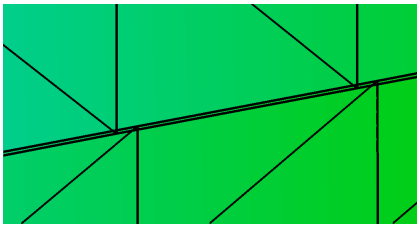
(a) Frictional coefficient $\mu = 0.19 < \tan\theta$: slip expected.

(b) Frictional coefficient $\mu = 0.21 > \tan\theta$: stick expected.

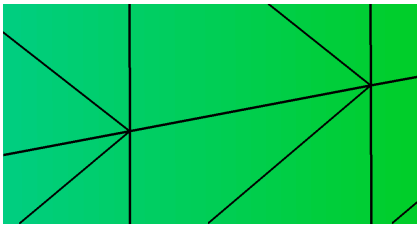
Figure 5: Horizontal displacement contours and deformed geometry for compressive loading of a plate with an inclined material interface using Nitsche's method. The inclination of the interface is such that $\tan\theta = 0.2$. Slip is predicted (left) when the frictional coefficient $\mu < \tan\theta$ while stick is predicted (right) when $\mu > \tan\theta$. The deformation is scaled by a factor of 2.



(a) Solution obtained using Penalty method when $\mu = 0.21$ and stick is expected



(b) Zoom of the mesh near the crack surface for Penalty method



(c) Zoom of the mesh near the crack surface for Nitsche's method

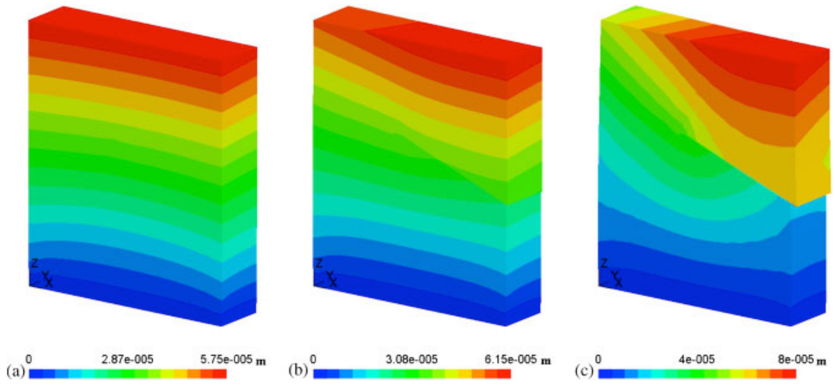
Figure 6: Comparison of accuracy in constraint enforcement between Nitsche's method and Penalty method when $\mu = 0.21$ and a stick state is expected. The deformation is scaled by a factor of 2.

clearly see continuous displacements when $\mu = 0.21 > \tan\theta$. Further, in Figure 6(a) we also plot the x-displacement contours obtained using penalty method with a normal penalty parameter $\alpha_N = 10^6$ and tangential penalty parameters $\alpha_\tau^1 = \alpha_\tau^2 = 10^4$ when $\mu = 0.21$. While we expect a stick state for this case, a slight discontinuity in the contours can be seen in the plot. We further highlight this by zooming into the mesh near the crack surface in Figure 6(b). This violation of constraints is a common characteristic of penalty regularization methods with the accuracy directly dependent on the regularization parameter. In theory, an essentially exact satisfaction of constraints is possible only in the limit when the regularization parameter is infinite. In practice, such large values result in ill-conditioned systems that manifest as spurious oscillations and stress-locking. Nitsche's method, on the other hand, results in a much more exact satisfaction of the constraint, for finite values of stabilization as can be seen from the zoom of the mesh plotted in Figure 6(c).

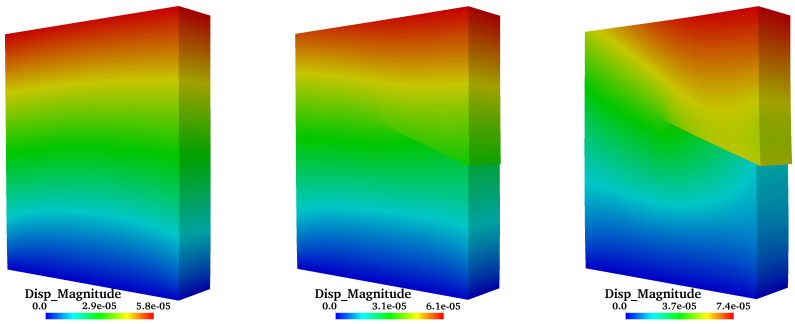
5.4. 3D planar crack under mixed mode loading

As a next example, we consider a 3D planar crack under mixed mode loading studied earlier in Gravouil *et al.* [65]. The problem geometry is similar to the one investigated in the study by Gravouil *et al.* [65] except that we consider a straight crack front. The material properties and loading conditions are also identical to the previous study (see Gravouil *et al.* [65]). We assume that Coulomb's frictional law exists at the crack interface with a coefficient of friction μ. We use an unstructured tetrahedral mesh with 5041 constant strain tetrahedral elements. The mesh aligns with the crack surface and is generated using Gmsh [71].

The specified tolerance of 10^{-16} was attained in thee, four and two iterations for $\mu = 0.0$, $\mu = 0.3$ and $\mu = 0.6$ respectively. The displacement magnitude contours from the current study and those presented in Gravouil *et al.* [65] are plotted in Figure 7 for comparison. As is evident from the plot, the obtained displacements are in excellent agreement. We also project the normal pressure to the x-y plane and then plot them as a function of x on the line $y = 0.0125$ in Figure 8. The y-range is set identical to that chosen in Gravouil *et al.* [65] for a fair comparison. From the Figure 8, we see that the normal pressures exhibit some oscillations which are increasing as the coefficient of friction decreases. However, unlike the oscillations reported for the standard LATIN algorithm in Gravouil *et al.* [65], the magnitude of the



(a) Displacement magnitude from Gravouil *et al.* [65]



(b) Displacement magnitude using Nitsche's method for $\mu = 0.6$

(c) Displacement magnitude using Nitsche's method for $\mu = 0.3$

(d) Displacement magnitude using Nitsche's method for $\mu = 0.0$

Figure 7: Comparison of displacement magnitude contours from Gravouil *et al.* [65] (top) and current study with Nitsche's method (bottom) for the 3D planar crack under mixed mode loading.

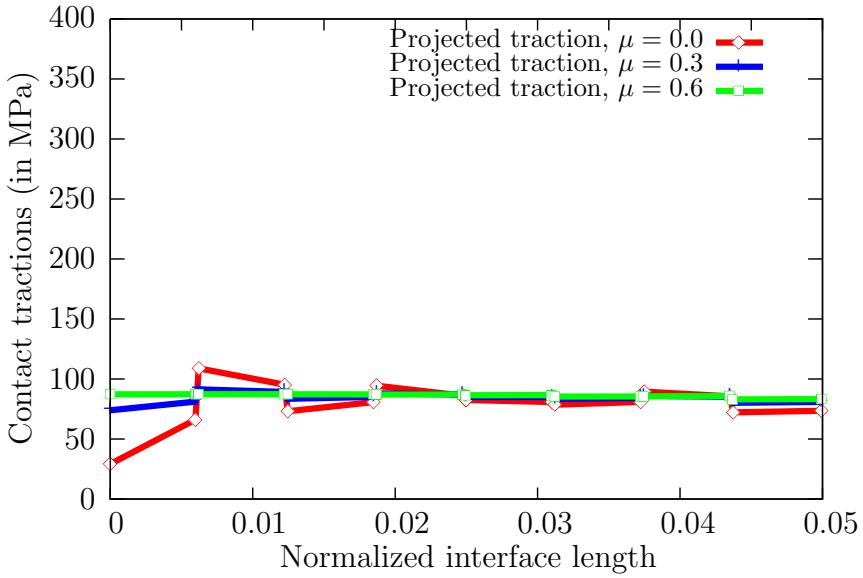


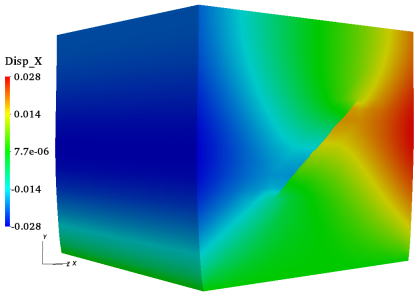
Figure 8: Projected tractions on the crack surface for the 3D planar crack under mixed mode loading.

oscillations reported here is small and could be related back to the poor stress approximation of linear tetrahedral elements as discussed in Section 5.2.

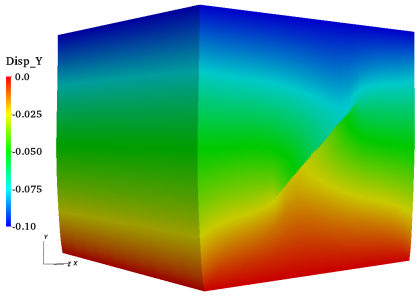
5.5. Inclined crack under compression

We now consider the example of frictional sliding on a pre-defined inclined crack under compressive loading. This example was earlier studied in plane-strain conditions by Dolbow *et al.* [32], Liu and Borja [26] and Annavarapu *et al* [63]. We extend this example to a three-dimensional setting and consider a pre-defined planar crack embedded in a unit cube. The crack is inclined at $\theta = 45$ degrees to the x-y plane and extends through the z-dimension. The material properties and boundary conditions are considered identical to previous studies (see Annavarapu *et al.* [63]). To reproduce plane-strain conditions, we also constrain the surfaces $z = 0$ and $z = 1$ in the z-direction. We consider a structured tetrahedral mesh with 20 divisions in each direction.

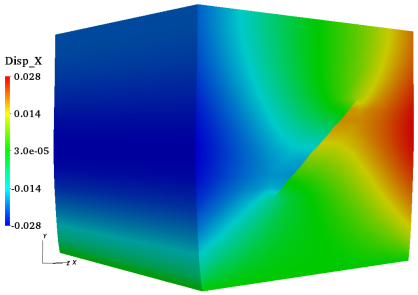
The x and y-displacement contours obtained using Nitsche's method are plotted in Figures 9(a)-(b). For comparison, we also plot the displacement contours using the penalty method. The penalty parameters are chosen as $\alpha_{\mathrm{N}} = \alpha_{\tau^1} = \alpha_{\tau^2} = 1.0 \times 10^{13}$ to remain consistent with earlier studies. The plots are in excellent agreement with earlier studies. The proposed method



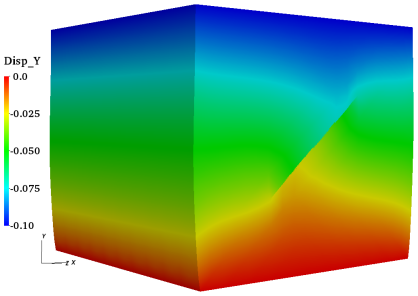
(a) x-displacement on using weighted Nitsche's method



(b) y-displacement on using weighted Nitsche's method



(c) y-displacement on using Penalty method with $\alpha_{\mathrm{N}} = \alpha_{\tau^1} = \alpha_{\tau^2} = 1.0 \times 10^{13}$



(d) y-displacement on using Penalty method with $\alpha_{\mathrm{N}} = \alpha_{\tau^1} = \alpha_{\tau^2} = 1.0 \times 10^{13}$

Figure 9: x and y displacement contours obtained using Nitsche's method (top) and Penalty method (bottom) for the crack face frictional contact problem.

converged in four Newton-Raphson iterations as opposed to the LATIN iterative strategy employed earlier in Dolbow *et al.* [32] which required in excess of 100 iterations to converge.

5.6. Frictional faults

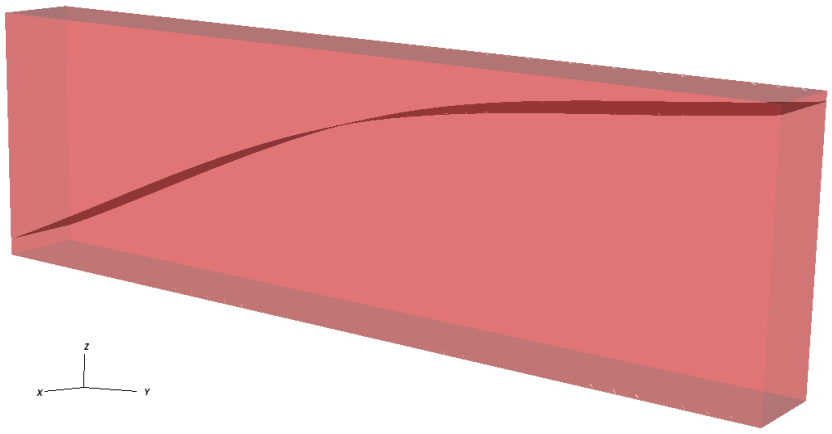
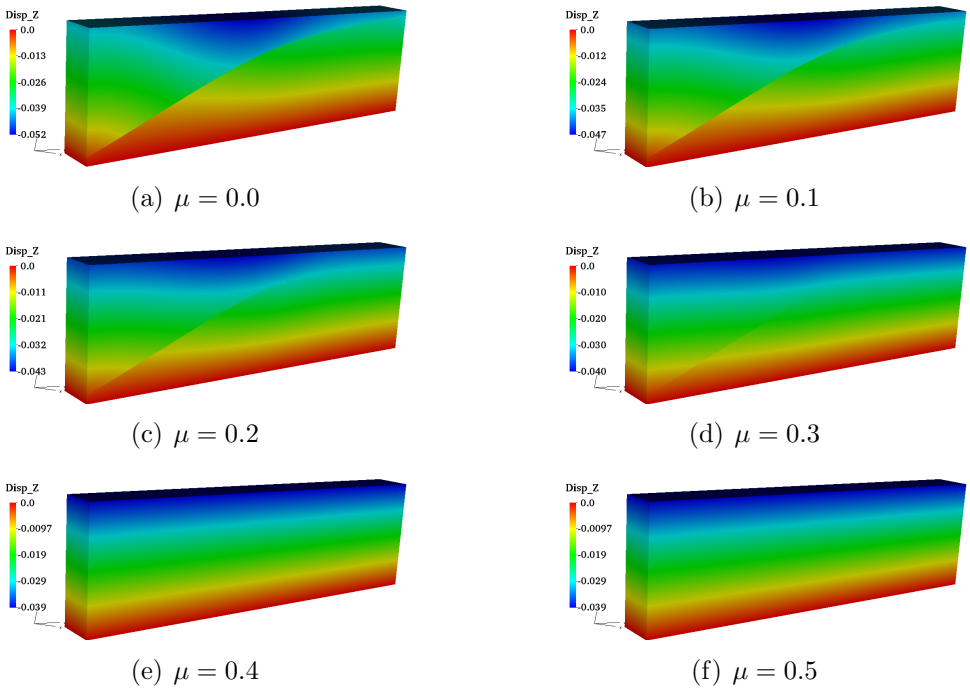


Figure 10: Illustration of the fault geometry for the active fault problem.

As a final example, we consider an application of the method to model frictional sliding along active faults. The example serves to test the method on curved interfaces and also demonstrates its utility in modeling a real world geological application. The example is motivated from an earlier study by Xing and Makinouchi [66] that conducted an explicit finite element calculation using penalty method for plate movement in the north-east zone of Japan. We consider the model to be a rectangular parallelepiped with dimensions of 1 km. in the x-direction, 10 kms. in the y-direction and 3 kms. in the z-direction. A curved fault surface extends through the geometry as shown in Figure 10 and represents a realistic fault geometry in the Pacific plate around Japan [66, 67]. Our objective in this study is to qualitatively study the effect of frictional coefficient of the fault interface on the relative slip along it. We ignore gravity effects in the current study. The model is constrained in all directions on the surface $z = 0$ km., constrained along x and y directions on the surfaces $y = 0$ km. and $y = 10$ km. and constrained along x-direction on the surfaces $x = 0$ km. and $x = 1$ km. A load of 100 MPa is applied on the surface $z = 3$ kms. as a traction boundary condition. We assume the material to have a Young's modulus of $E = 44.8$ GPa and

a Poisson's ratio of $\nu = 0.3$. Coulomb's frictional behavior is assumed to hold for the fault interface and we run the simulation for five different values ranging from $\mu = 0.0$ and $\mu = 0.5$. The model is meshed with 11894 constant strain tetrahedral elements conforming with the fault geometry.



(a) $\mu = 0.0$ (b) $\mu = 0.1$

(c) $\mu = 0.2$ (d) $\mu = 0.3$

(e) $\mu = 0.4$ (f) $\mu = 0.5$

Figure 11: z-displacement contours obtained using Nitsche's method with different coefficients of friction for the active frictional fault problem.

The results of this study are plotted in Figure 11. As one would expect, a smaller value of frictional coefficient increases the tendency of the fault surfaces to slip relative to one another which is clearly evident in the plots as larger discontinuities. These predictions are also in close agreement with the earlier study by Xing and Makinouchi [66]. Further, we also tabulate the Newton-Raphson convergence behavior of Nitsche's method for various coefficients of frictions in Table 2. The asymptotically quadratic rates of convergence of Newton's method are evident for lower values the coefficient of friction μ while for $\mu = 0.4$ and $\mu = 0.5$, the method converges in two iterations indicating no slip on the entire fault surface which is also supported by the continuous displacement contours in Figures 11(e) and 11(f).

Table 2: Newton-Raphson convergence behaviour of Nitsche's method for the frictional fault problem.

Iteration number	Energy norm					
	$\mu = 0.0$	$\mu = 0.1$	$\mu = 0.2$	$\mu = 0.3$	$\mu = 0.4$	$\mu = 0.5$
1	1.00e+00	1.00e+00	1.00e+00	1.00e+00	1.00e+00	1.00e+00
2	2.00e-01	5.72e-02	1.45e-02	1.47e-02	3.17e-28	3.17e-28
3	4.07e-28	2.60e-03	3.05e-04	1.60e-05		
4		3.86e-09	8.78e-05	2.95e-06		
5		1.38e-18	1.63e-05	2.41e-07		
6			3.42e-06	1.10e-08		
7			2.80e-07	9.78e-13		
8			1.01e-08	1.52e-24		
9			5.34e-11			
10			1.71e-20			

6. Conclusion

We proposed a Nitsche stabilized approach to model frictional contact on three-dimensional crack surfaces. We briefly reintroduced the variational form associated with this method and recalled the weighted approach used for the Nitsche consistency terms. We also provided algorithmic details associated with the derivation of contact tangent operator in the presence of Nitsche consistency terms. Several numerical examples demonstrated the efficiency of the method for frictional contact applications for both linear tetrahedral and trilinear hexahedral elements.

The numerical examples demonstrated that the method is robust and does not suffer from over or under-penalization like the penalty methods. Unlike augmented Lagrange multiplier methods, the formulation is purely displacement based and hence results in a smaller system matrix and eliminates augmentation loops. Finally, the quadratic convergence of Newton-Raphson iterative schemes is preserved. While we confined attention to mesh-aligned interfaces in the present work, the method also applies to X-FEM interfaces as demonstrated for a two-dimensional case before [63, 64]. With these considerations, from a numerical standpoint, we contend that the method is much more efficient than augmented Lagrange multiplier or penalty ap-

proaches for elastic contact and small sliding problems on both mesh-aligned and X-FEM type interfaces.

Going forward, several possible extensions of the method are of interest. Sharp algebraic estimates for the Nitsche stabilization parameter for quadrilateral and hexahedral elements remain to be developed. The extension of the approach to large sliding at the interface could be useful. Investigating more general constitutive behavior at the interface is also of interest. Finally, we mention the extension of the approach to incorporate bulk nonlinearities as a possible future avenue of research.

7. Acknowledgements

This document was prepared by LLNL under Contract DE-AC52-07NA27344. The authors gratefully acknowledge this support.

References

[1] Ladeveze, P. *Nonlinear Computational Structural Mechanics*. New York: Springer-Verlag, 1998.

[2] Ribeaucourt, R., Baietto, M. C., Gravouil A., A new fatigue frictional contact crack propagation model with the coupled X-FEM/LATIN method, Comput. Meth. Appl. Mech. Eng., **196**(33-34):3230-3247, 2007.

[3] Ling, H.I., Cheng, A.,H. -D., Rock sliding induced by seismic force, Int. J. Rock Mech. Min. Sci., **34**(6):1021-1029, 1997.

[4] Bahaaddini, M., Sharrock, G., Hebblewhite, B. K., Numerical direct shear tests to model shear behavior of rock joints, Comput. Geotech., **51**:101-115, 2013.

[5] Seo, Y.S., Geong, G.C., Kim, J.S., Ichikawa, Y., Microscopic stress observation and contact stress analysis of granite under compression, Eng. Geology, **63**(3-4):259-275, 2002.

[6] Scholtés, L., Donzé, F., Modeling progressive failure in fractured rock masses using a 3D discrete element method, Int. J. Rock Mech. Min. Sci., **52**:18-30, 2012.

[7] Gehle, C., Kutter, H.K., Breakage and shear behavior of intermittent rock joints, Int. J. Rock Mech. Min. Sci., **40**(5):687-700, 2003.

[8] Min, K., Jing, L., Numerical determination of equivalent elastic compliance tensor for fractured rock masses using the distinct element method, Int. J. Rock Mech. Min. Sci., **40**(6):795-816, 2003.

[9] Simo, J. C., Oliver, J., Armero, F., An analysis of strong discontinuities induced by strain softening in rate independent inelastic solids, Comput. Mech., **12**:277-296, 1993.

[10] Xu, P., Needleman, A., Numerical simulations of fast crack growth in brittle solids, J. Mech. Phy. Solids, **42**(9):1397-1434, 1994.

[11] Perić, D., Owen, D.R.J., Computational model for 3-D contact problems with friction based on the penalty method, Int. J. Num. Meth. Eng., **35**(6):1289-1309, 1992.

[12] Settgast, R.R., Johnson, S.M., Fu, P., Walsh, S.D.C., Annavarapu, C., Hao, Y., White, J.A., Ryerson, F.J., GEOS: A framework for massively parallel multi-physics simulations. Theory and Implementation., LLNL Technical Report 654611, 2014

[13] Camacho, G.T., Ortiz, M., Computational modelling of impact damage in brittle materials, Int. J. Solids Struct., **33**(20–22):2899-2938, 1996.

[14] Settgast R.R., Rashid, M.M., Continuum coupled cohesive zone elements for analysis of fracture in solid bodies, Eng. Fracture Mech., **76**(11):1614-1635, 2009.

[15] Radovitzky, R., Seagreaves, A., Tupek, M, Noels, L., A scalable 3D fracture and fragmentation algorithm based on a hybrid, discontinuous Galerkin, cohesive element method, Comput. Meth. Appl. Mech. Eng., **200**(1-4):326–344, 2012.

[16] Moës, N., Dolbow, J., Belytschko, T., A finite element method for crack growth without remeshing, Int. J. Num. Meth. Eng., **46**(1):131-150, 1999.

[17] Duarte C.A., Babŭska, I., Oden, J.T., Generalized finite element methods for three-dimensional structural mechanics problems, Comput. Struct., **77**(2):215-232, 2000.

[18] Gonçalves, J.P.M., deMoura, M.F.S.F., de Castro, P.M.S.T., Marques, A.T., Interface element including point-to-surface constraints for three-dimensional problems with damage propagation, Eng. Comput., **17**(1):28-47, 2000.

[19] Day, R. A., Potts, D. M., Zero thickness interface elements numerical stability and application, Int. J. Num. Anal. Meth. Geomech., **18**:689-708, 1994.

[20] Schellekens J.J.C., De Borst, R., On the numerical integration of interface elements, Int. J. Num. Meth. Eng., **36**(1):43-66, 1993.

[21] Warner, D.H., Molinari, J.F., Micromechanical finite element modeling of compressive fracture in confined alumina ceramic, Acta Materialia, **54**(19):5135-5145, 2006

[22] Espinosa, H., Zavattieri, P., A grain level model for the study of failure initiation and evolution in polycrystalline brittle materials. Part I: Theory and numerical implementation, Mech. Mat., **35**(3-6), 333-364, 2003.

[23] Espinosa, H., Zavattieri, P., A grain level model for the study of failure initiation and evolution in polycrystalline brittle materials. Part II: Numerical Examples, Mech. Mat., **35**(3-6), 365-394, 2003.

[24] An, T., Qin, F., Intergranular cracking simulation of the intermetallic compound layer in solder joints, Comput. Materi. Sci., **79**:1-14, 2013.

[25] Ladeveze P., Nouy, A., Loiseau, O., A multiscale computational approach for contact problems, Comput. Meth. Appl. Mech. Engrg., **191**(43):4869–4891, 2002.

[26] Liu, F. and Borja, R.I., A contact algorithm for frictional crack propagation with the extended finite element method, Int. J. Num. Meth. Eng., **76**(10):1489–1512, 2008.

[27] Liu, F. and Borja, R.I., An extended finite element framework for slow-rate frictional faulting with bulk plasticity and variable friction, Int. J. Num. Anal. Meth. Geomech., **33**(13): 1535–1560, 2009.

[28] Liu, F. and Borja, R.I., Finite deformation formulation for embedded frictional crack with the extended finite element method, Int. J. Num. Meth. Eng., **82**(6):773–804, 2010.

[29] Liu, F. and Borja, R.I., Stabilized low-order finite elements for frictional contact with the extended finite element method, Comput. Meth. Appl. Mech. Engrg., **199**(37-40):2456–2471, 2010.

[30] Armero, F. and Petocz, E. A new dissipative time-stepping algorithm for frictional contact problems: formulation and analysis, Comput. Meth. Appl. Mech. Engrg., **179**:151–178, 1999.

[31] Nistor, I., Guiton, M.L.E., Massin, P., Moës, N., and Geniaut, S., An X-FEM approach for large sliding contact along discontinuities, Int. J. Num. Meth. Eng., **78**(12):1407–1435, 2009.

[32] Dolbow, J. E., Möes, N. and Belytschko, T., An extended finite element method for modeling crack growth with frictional contact, Comput. Meth. Appl. Mech. Engrg., **190**(51-52):6825–6846, 2001.

[33] Simo, J. C. and Laursen, T. A., An augmented Lagrangian treatment of contact problems involving friction, Comput. Struct., **42**(1):97–116, 1992.

[34] Kim, T. Y., Dolbow J. E. and Laursen, T. A., A mortared finite element method for frictional contact on arbitrary interfaces, Comput. Mech., **39**(3):223–235, 2007.

[35] Khoei, A.R. and Mousavi, S.M.T., Modeling of large-deformation - large sliding contact via the penalty X-FEM technique, Comp. Mat. Sci., **48**(3):471-480, 2010.

[36] Nitsche, J., Uber ein variationsprinzip zur losung von Dirichlet-problemen bei verwendung von teilraumen, die keinen randbedingungen unterworfen sind, Anh. Math. Sem. Univ. Hamburg, **36**:9–15, 1970.

[37] Wriggers, P., and Zavarise, G., A formulation for frictionless contact problems using a weak form introduced by Nitsche, Comput. Mech., **41**(3):407–420, 2008.

[38] Giner, E., Tur, M., Tarancón, J.E., Fuenmayor, F.J., Crack face contact in X-FEM using a segment-to-segment approach, Int. J. Num. Meth. Eng., **82**:1424-1449, 2010.

[39] Mueller-Hoppe, D.S., Wriggers, P., Loehnert, S., Crack face contact for a hexahedral-based X-FEM formulation, Comput. Mech., **49**:725-734, 2012.

[40] Chouly, F., Hild, P., Renard, Y., Symmetric and non-symmetric variants of Nitsche's method for contact problems in elasticity: theory and numerical experiments, submitted for publication.

[41] Renard, Y., Generalized Newton's methods for the approximation and resolution of frictional contact problems in elasticity, Comput. Meth. Appl. Mech. Engrg., **256**:38-55, 2013.

[42] Chouly, F., Hild, P., A Nitsche-based method for unilateral contact problems: Numerical Analysis, Siam J. Num. Analy., **51**(2):1295-1307, 2013.

[43] Chouly, F., An adaptation of Nitsche's method to the Tresca friction problem, J. Math. Analy. Appl., **411**(1):329-339, 2014.

[44] Hild, P., Renard, Y., Stabilized lagrange multiplier method for the finite element approximation of contact problems in elastostatics, Numer. Math., **15**(1):101-129, 2010.

[45] Masud, A., Truster, T.J., Bergman, L.A., A unified formulation for interface coupling and frictional contact modeling with embedded error estimation, Int. J. Num. Meth. Eng., **92**(2):141–177, 2012.

[46] Truster, T.J., Eriten, M., Polycarpou, A.A., Bergman, L.A., Masud, A., Stabilized interface methods for mechanical joints: Physics-based models and variationally consistent embedding, Int. J. Solids, Struct., **50**(14-15):2132-2150, 2013.

[47] Coon, E.T., Shaw, B.E., and Spiegelman, M., A Nitsche-extended finite element method for earthquake rupture on complex fault systems, Comput. Meth. Appl. Mech. Engrg., **200**:2859–2870, 2011.

[48] Simone, A., Partition of unity-based discontinuous elements for interface phenomena: computational issues, **20**(6):465-478, 2004.

[49] Siavelis, M., Guiton, M. L. E., Massin, P., Moës, N., Large sliding contact along branched discontinuities with X-FEM, Comput.Mech., **52**(1):201-219, 2013.

[50] Brezzi, F., Existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers, RAIRO Oper-Res., **8**(2):129-151, 1974.

[51] Wohlmuth, B. I., A mortar finite element method using dual spaces for the Lagrange Multiplier, SIAM J. Numer. Anal., **38**(3):989-1012, 2000.

[52] Embar, A., Dolbow, J., Harari, I., Imposing Dirichlet boundary conditions with Nitsche's method and spline-based finite elements, Int. J. Num. Meth. Eng., **83**(7):877-898, 2010.

[53] Harari, I., Shavelzon, E., Embedded kinematic boundary conditions for thin plate bending by Nitsche's approach, Int. J. Num. Meth. Eng., **92**(1):99-114, 2012.

[54] Sanders, J., Puso, M., Laursen, T., A Nitsche embedded mesh method, Comput. Mech. **49**(2):243-257, 2012.

[55] Hautefeuille, M., Annavarapu, C., Dolbow, J., Robust imposition of Dirichlet boundary conditions on embedded surfaces, Int. J. Num. Meth. Eng., **90**(1):40–64, 2012.

[56] Béchet, E., Moës, N., Wohlmuth, B., A stable Lagrange multiplier space for stiff interface conditions within the extended finite element method, Int. J. Num. Meth. Eng., **78**(8):931–954, 2012.

[57] Khoei, A. R., Nikbakht, M., An enriched finite element algorithm for numerical computation of contact friction problems, Int. J. Mech. Sci., **49**:183–199, 2007.

[58] Khoei, A. R., Shamloo, A., Azami, A. R., Extended finite element method in plasticity forming of powder compaction with contact friction, Int. J. Solids. Struct., **43**:5421–5448, 2006.

[59] Wriggers, P., Vu Van, T. and Stein, E. Finite element formulation of large deformation impact-contact problems with friction, Comput. Struct., **37**(3):319–331, 1990.

[60] Oden, J. T. and Pires, E. B., Algorithms and numerical results for finite element approximations of contact problems with non-classical friction laws, Comput. Struct., **19**(1-2):137–147, 1983.

[61] Annavarapu, C., Hautefeuille, M and Dolbow, J. E., A robust Nitsche's formulation for interface problems, Comput. Meth. Appl. Mech. Engrg., **225-228**(4):44–54, 2012.

[62] Annavarapu, C., Hautefeuille, M and Dolbow, J. E., Stable imposition of stiff constraints in explicit dynamics for embedded finite element methods, Int. J. Num. Meth. Eng., **92**(2):206–228, 2012.

[63] Annavarapu, C., Hautefeuille, M. and Dolbow, J. E., A Nitsche stabilized finite element method for frictional sliding on embedded interfaces. Part I: Single interface, Comput. Meth. Appl. Mech. Engrg., **268**(1): 417–436, 2014.

[64] Annavarapu, C., Hautefeuille, M. and Dolbow, J. E., A Nitsche stabilized finite element method for frictional sliding on embedded interfaces. Part II: Intersecting interfaces, Comput. Meth. Appl. Mech. Engrg., **267**(1): 318–341, 2013.

[65] Gravouil, A., Pierres, E. and Baietto, M. C., Stabilized global–local X-FEM for 3D non-planar frictional crack using relevant meshes, Int. J. Num. Meth. Eng., **88**:1449–1475, 2011.

[66] Xing, H. L., Makinouchi, A., Finite element modeling of multi-body contact and its application to active faults, Concurrency Computat.: Pract. Exper., 2002.

[67] Kanai, T., Makinouchi, A., Oishi, Y. Development of tectonic CAD/Database systems, Abstracts of the International Workshop on Solid Earth Simulation and ACES WG Meeting, Mitsu'ura, M. et al.(eds.), Tokyo, January 17–21, 2000.

[68] Simo, J. C. and Hughes, T. J. R., *Computational Inelasticity*, Springer, 2000.

[69] Laursen, T. A., *Computational Contact and Impact Mechanics*, Springer, 2002.

[70] Wriggers, P., *Computational Contact Mechanics*, Second Edition, Springer, 2006.

[71] Geuzaine, C., Remacle, J-F, Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities, Int. J. Num. Meth. Eng., **79**(11):1309-1331, 2009

[72] Oliver, J., Hartmann, S., Cante, J. C., Weyler, R. and Hernández, J. A., A contact domain method for large deformation frictional contact problems. Part 1: theoretical basis, Comput. Meth. Appl. Mech. Engrg., **198**(33-36): 2591–2606, 2009.

[73] Hartmann, S., Oliver, J., Cante, J. C. and Weyler, R. and Hernández, J. A., A contact domain method for large deformation frictional contact problems. Part 2: Numerical aspects, Comput. Meth. Appl. Mech. Engrg., **198**(33-36): 2607–2631, 2009.

ARMA 14-7404

A Stabilized Finite Element Approach to Model Contact Conditions in Fractured Subsurface Media

Chandrasekhar Annavarapu, Randolph Settgast, Scott Johnson, Pengcheng Fu and Eric B. Herbold
Lawrence Livermore National Laboratory, Livermore, California, United States of America

This paper was prepared for presentation at the 48th US Rock Mechanics / Geomechanics Symposium held in Minneapolis, MN, USA, 1-4 June 2014.

ABSTRACT: We propose a stabilized approach based on Nitsche's method for enforcing contact constraints over crack surfaces. The proposed method addresses the shortcomings of conventional penalty and augmented Lagrange multiplier approaches by combining their attractive features. Similar to an augmented Lagrange multiplier approach, the proposed method has a consistent variational basis resulting in stronger enforcement of the non-interpenetration constraint. At the same time, the proposed method is purely displacement-based and alleviates the stability challenges common to mixed methods. The method also retains the computational efficiency of penalty approaches by eliminating the outer augmentation loop necessary for augmented Lagrangian approaches and resulting in smaller system matrices.

1. INTRODUCTION

The transient mechanical response of rock surfaces in contact is of prime importance in many engineering applications ranging from crack-closure effects in micro-fractures to frictional sliding between joint or fault surfaces. These effects span length-scales that are orders of magnitude apart yet are equally significant at either end of this spectrum. Robust numerical strategies that enable modeling of these effects can have far-reaching consequences in guiding important engineering decisions such as seismic hazard characterization and engineering fracture networks for efficient extraction of shale gas and geothermal energy [1-6].

Fracture problems in a Lagrangian framework typically require a constantly changing mesh topology and the ability to deal with discontinuities both of which are significant challenges for the finite element method. A variety of approaches have been proposed to address these issues including strong discontinuity approaches [7] interface element techniques [8], discontinuous Galerkin methods [9] and extended finite element methods [10]. However, most of these methods suffer from numerical instabilities when confronted with the fundamental problem of crack closure [11, 12] that is hard to neglect for geomaterials and rocks. In this paper, we present an approach that alleviates these numerical concerns and highlight its efficiency with several numerical examples.

The key numerical issue in resolving crack closure effects concerns the enforcement of nonlinear contact constraints. The most commonly used approaches, *viz.* the penalty and variants of Lagrange multiplier methods for enforcing contact conditions either suffer from lack of accuracy in the enforcement of the non-interpenetration condition or from spurious oscillations in contact stresses [13, 14]. We present an alternative stabilized approach based on Nitsche's method [15] for this class of problems. The method combines the attractive features of penalty and Lagrange multiplier approaches to yield a robust and computationally efficient alternative. The method previously demonstrated for frictional contact in X-FEM for constant strain triangles by Annavarapu et al. [16, 17] is extended here for bilinear quadrilaterals and three-dimensional problems for crack faces that lie along inter-element boundaries. Here, we confine attention to perfect contact and frictionless sliding.

The rest of the paper is organized as follows. In Section 2, we describe the governing equations and the theoretical framework for the proposed approach. In Section 3, we consider several benchmark examples to illustrate the performance of the method. Finally in

Section 4, we offer concluding remarks and an outlook for the work.

2. GOVERNING EQUATIONS

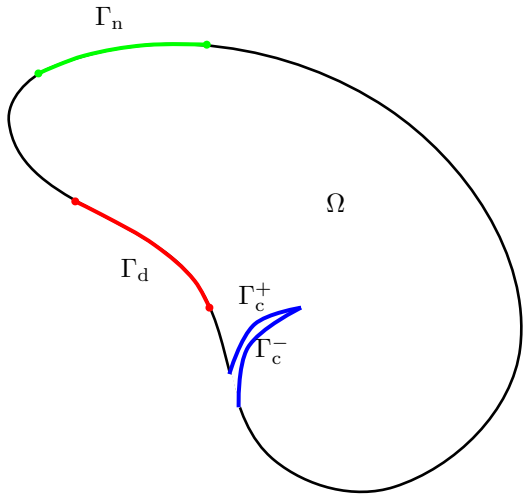


Fig. 1. Notation for the model problem. Domain Ω, the Dirichlet boundary Γ_d and the Neumann boundary Γ_n are as shown. The complementary part of the boundary is traction free. The normal to the boundary of a domain is considered to point outwards from the domain.

We begin by considering a domain Ω and its boundary Γ as shown in Figure 1. Further, we consider Γ_c to represent a crack surface with Γ_c^- and Γ_c^+ representing the initially coincident crack faces. The governing equations for small deformation elastostatics are now given in indicial notation as:

$$\begin{aligned} \sigma_{ij,j} &= 0 \quad \text{in } \Omega, \\ u_i &= u_i^d \text{ on } \Gamma_d, \\ \sigma_{ij} n_j &= h_i \text{ on } \Gamma_n, \end{aligned} \tag{1}$$

where σ_{ij} and u_i denote the components of the stress and displacement fields in domain Ω, respectively, and n_j the components of the unit outward normal. The displacement is fixed to the prescribed field u^d on the Dirichlet portion of the boundary, and h_i denotes the prescribed traction on the Neumann portion of the boundary. We assume a linear elastic response for the constitutive relationship in the bulk domain. With respect to the constitutive relationship at the crack surface, we develop the proposed approach for perfectly tied crack surfaces and for crack surfaces under small-deformation frictionless sliding. We relate the tractions, t_i^+ and t_i^- from both sides of the crack surface through a force-balance relation. Additionally, for convenience, we define a single traction field, t_i on the $n^+ - \tau^+$ plane of the crack face Γ_c^+ such that:

$$t_i = t_i^+ = -t_i^- \text{ on } \Gamma_c. \tag{2}$$

Furthermore, the traction field and the displacements on the interface can now be expressed in the normal and tangential planes along the interface as:

$$\begin{aligned} t_i &= t_N n_i^+ + t_\tau \tau_i^+ \\ u_i^m &= u_N^m n_i^+ + u_\tau^m \tau_i^+; \quad m = +,- \end{aligned} \tag{3}$$

Now in case of perfect contact, in addition to the traction continuity, we also have displacement continuity across the crack surface such that:

$$u_N^+ = u_N^-; \quad u_\tau^+ = u_\tau^-. \tag{4}$$

For frictionless sliding behavior, the continuity of displacements in the tangential direction no longer applies. In addition, owing to the lack of friction, the crack surfaces develop no stresses in the tangential direction *i .e.* $t_\tau = 0$.

2.1. Variational Formulation

The variational form of the governing equations described above can be derived as: Find $u_i \in U_i$ such that:

$$\int_\Omega w_{(i,j)} \sigma_{ij} \, d\Omega - \int_{\Gamma_c^+} w_i^+ t_i^+ \, d\Gamma - \int_{\Gamma_c^-} w_i^- t_i^- \, d\Gamma = \int_{\Gamma_n} w_i h_i \, d\Gamma \tag{5}$$

for all $w_i \in V_i$ where U_i and V_i are spaces of sufficiently smooth functions for the displacements and variations respectively. From the traction continuity equation (2), we have: Find $u_i \in U_i$ such that:

$$\int_\Omega w_{(i,j)} \sigma_{ij} \, d\Omega - \int_{\Gamma_c} [[w_i]] t_i \, d\Gamma = \int_{\Gamma_n} w_i h_i \, d\Gamma \tag{6}$$

for all $w_i \in V_i$ where $[[w_i]]$ is the jump in the variations across the crack face. The first and third terms in equation (6) are standard. We now take a closer look at the second term that represents the contact virtual work. From equation (3), we write the contact integral as:

$$\int_{\Gamma_c} [[w_i]] t_i \, d\Gamma = \int_{\Gamma_c} ([[w_N]] t_N + [[w_\tau]] t_\tau) \, d\Gamma \tag{7}$$

For Nitsche's method, the traction on the crack surface is defined in terms of the bulk stresses and interpenetration such that:

$$t_N = n_i^+ \langle \sigma_{ij} \rangle_\gamma n_j^+ - \alpha_N [[u_N]],$$

$$t_\tau = \begin{cases} \tau_i^+ \langle \sigma_{\{ij\}} \rangle_\gamma n_j^+ - \alpha_\tau [[u_\tau]] & \text{for perfect contact} \\ 0 & \text{frictionless contact} \end{cases}$$

where $\langle \sigma_{ij} \rangle_\gamma = \gamma^1 \sigma_{ij}^+ + \gamma^2 \sigma_{ij}^-$ represents the weighted average traction across the crack face. The formulation is consistent for all weights $\gamma^1 > 0$ and $\gamma^2 > 0$ such that $\gamma^1 + \gamma^2 = 1$. The terms $[[u_N]]$ and $[[u_\tau]]$ represent the normal and tangential gap respectively while α_N and α_τ represent the stabilization parameters in the normal and tangential directions. The stabilization and weighting parameters that appear in the proposed approach are chosen such that the discrete system of equations remains positive definite and well-conditioned. For conciseness, we omit the details concerning the precise definitions of these parameters and refer the interested reader to Annavarapu [18] for additional details.

Though the stabilization terms in the proposed formulation appear similar in form to the penalization terms in the penalty method, it should be emphasized that the stabilization introduced here is from a purely numerical perspective and has little bearing on the satisfaction of the non-interpenetration constraint. In fact, the consistency of the formulation is ensured for any non-zero value of the stabilization parameter. Finally, we remark that in the presented form, the formulation is non-symmetric. Interfacial sliding behavior is often characterized by coupling between normal and shear directions (for e.g. Coulomb's law) and this coupling manifests as the non-symmetry of the consistent tangent matrix and consequently we leave out the symmetry terms. However, for perfect and frictionless contact, where there is no coupling between normal and shear directions it is trivial to recover the symmetry of the formulation by adding the conjugate terms to equation (7).

3. NUMERICAL EXAMPLES

In this section, we consider several benchmark examples to validate the performance of the proposed method. We compare the results of the proposed approach with those of the penalty method and highlight its advantages.

3.1. Contact Patch Test

In contact mechanics, the effectiveness of a method in enforcing contact constraints is often tested by means of a patch test [19] The main idea behind this test is to examine the performance of the numerical approach in reproducing uniform strain conditions. The domain of interest considered here is a 1.0 X 1.0 square region. The material within the square block is considered to obey linear elastic constitutive behavior and has a Young's modulus of E = 1.0 Pa and a Poisson's ratio of = 0.0. The surface y = 0 is constrained to move in the y-direction and the point (x,y) = (0,0) is constrained to move in both x and y-directions. A uniform stretch of u_y = 0.01 m. is applied on the surface y = 1. It is easy to verify that for the specified loading and boundary conditions, an analytical expression for the displacement field can be derived and is given as:

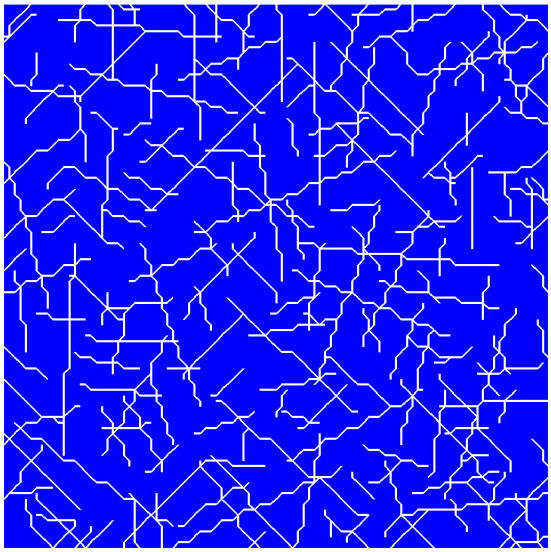


Fig. 2. Random fracture network distribution in a square elastic block.

$$\begin{aligned} u_x &= 0 \\ u_y &= 0.01y \end{aligned} \tag{8}$$

Further, a random fracture network is distributed across the domain as shown in Figure 2. In addition, we assume perfect contact conditions exist across every crack face. The results of this study obtained from Nitsche's and penalty approaches are presented in Figures 3(b) and 3(c) respectively. In these Figures, we plot the displacements in the y-direction and magnify them by a factor of 100. The penalty parameter prescribed for the penalty method in both the normal and tangential directions is chosen identically as E/h = 100.0 units where E is the Young's modulus and h is the characteristic mesh size. For Nitsche's method, since the method is consistent for any non-zero value of the stabilization parameter, the stabilization parameter is set to 1 x 10^{-4} in both the normal and tangential directions.

It is clear from the results that Nitsche's method essentially enforces the contact conditions in an exact sense even with little stabilization. Penalty method, on the other hand, exhibits poor accuracy for finite penalties. While the accuracy in constraint enforcement can be improved upon by tuning the penalty parameters, these result in ill-conditioned systems and spurious oscillations in contact pressures as shown later in this Section.

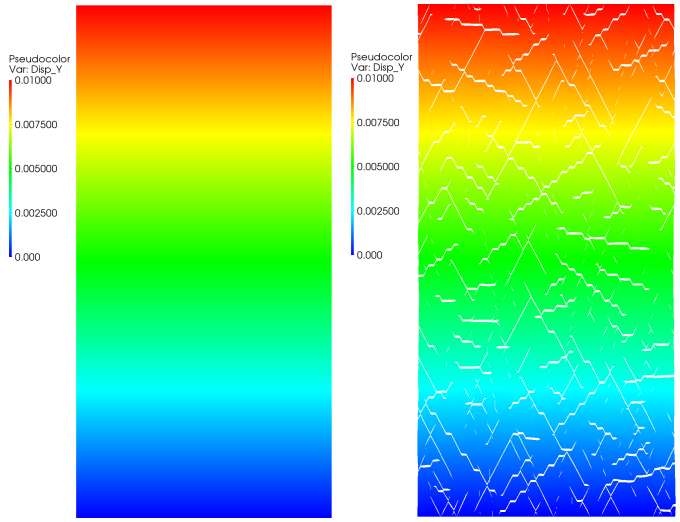


Fig. 3. Displacement contours in the y-direction for the contact patch test obtained using Nitsche's method (left) and the penalty method (right). The plotted displacements are magnified by a factor of 100.

3.2. *Plane-strain Frictionless Sliding contact*

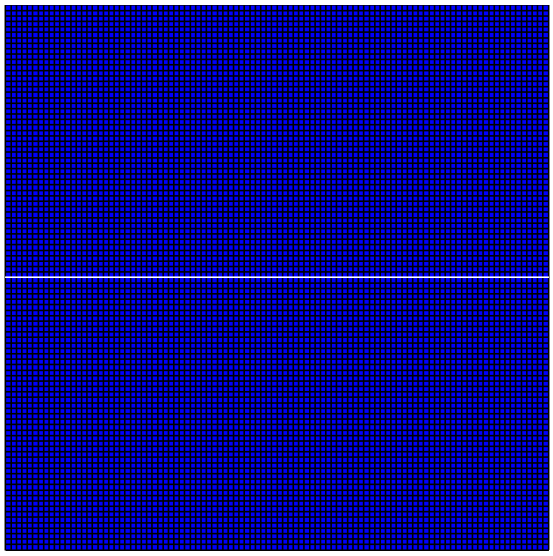


Fig. 4. Mesh and crack geometry for the plane-strain frictionless sliding example

We now revisit a horizontal crack in frictionless sliding earlier investigated by Liu and Borja [12]. Here we consider a square elastic block of unit length with a horizontal crack extending through the middle as shown in Figure 4. The material is considered to have a Young's modulus of E = 10 GPa and a Poisson's ratio ν = 0.31 for the material above the crack and ν = 0.29 for material below crack surface. The slight difference in Poisson's ratio provides conditions for sliding when the material is compressed. Further, we assume the bulk material obeys linear elasticity and deforms under plane-strain conditions. The boundary conditions are such that both the top and bottom surfaces are clamped laterally while the bottom surface is clamped vertically as well. Further, a uniform displacement of u_y = -0.1 m. is applied at the top surface.

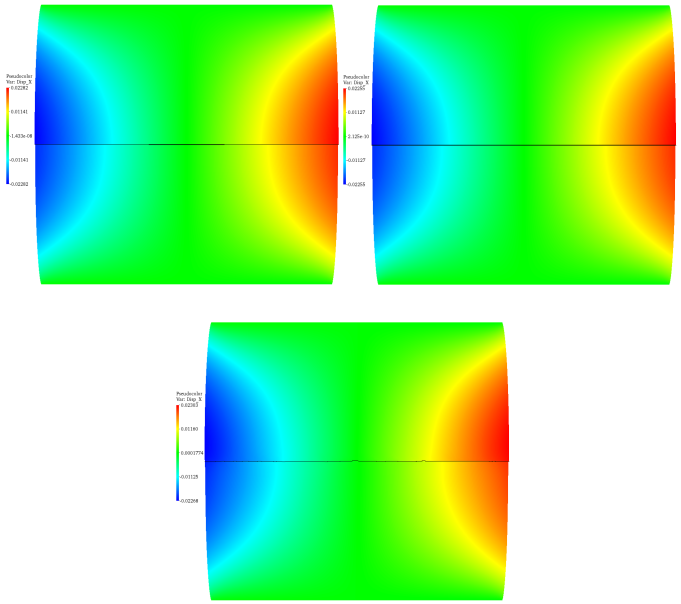


Fig. 5. X-displacement contours obtained using Nitsche's method (top left), penalty method with a finite penalty (top right) and with a large penalty (bottom). The colours range from -0.02 (blue) to 0.02 (red).

In Figure 5, we plot displacement contours in the x-direction obtained using Nitsche's method and the penalty method for two different penalty parameters. Our choice for the penalty parameter here is illustrative of the two extremes (a) a large value of $\alpha_N = 10^{12}$ GPa/m that enforces the non-interpenetration constraint with high accuracy and (b) a finite value of $\alpha_N = 10^3$ GPa/m that ensures good scaling and well-conditioned system matrices but poor constraint enforcement. It should be noted that the displacement contours are nearly indistinguishable for all three cases. The differences become apparent by comparing the stress and interpenetration at the interface. In Figure 6, we plot σ_{yy} obtained using Nitsche's method and the penalty method with a penalty parameter of $\alpha_N = 10^{12}$ GPa/m. The severe stress locking exhibited in the results obtained using the penalty method is evident while the stress obtained using Nitsche's method is smooth. A smaller penalty of $\alpha_N = 10^3$ GPa/m returns smooth stresses but the interpenetration is much higher as seen in Figure 7. This example illustrates the fundamental difficulty that numerical analysts face in choosing the ``right'' value for the penalty parameter.

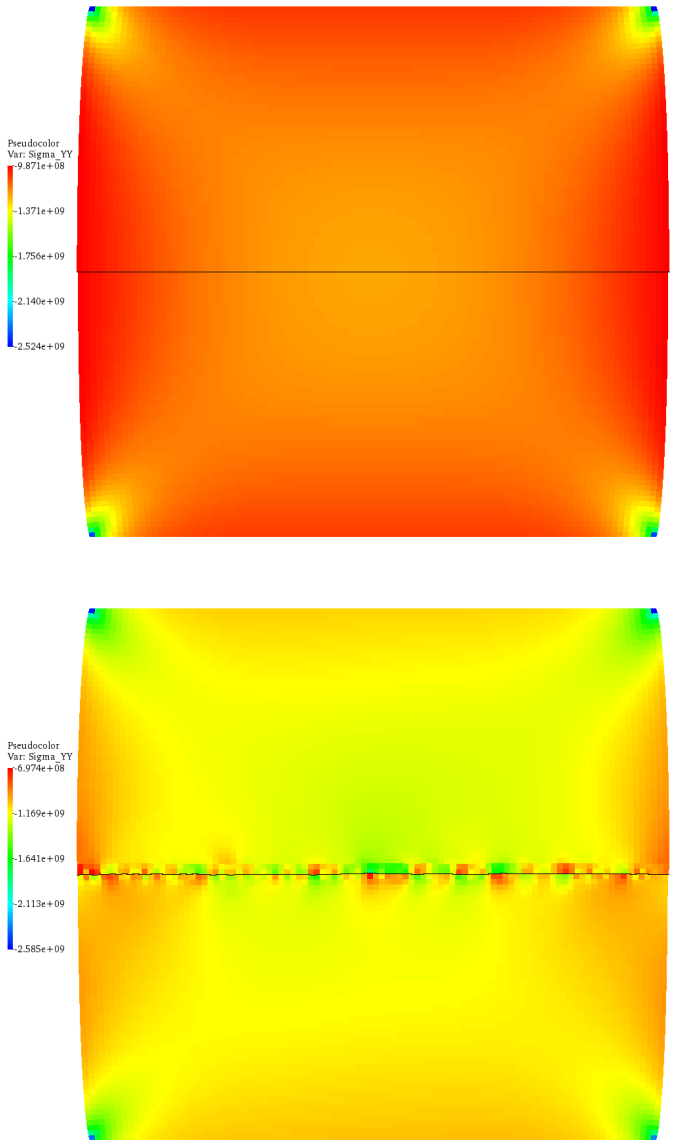


Fig. 6. Contours of σ_{yy} obtained using Nitsche's method (top) and the penalty method with a large penalty (bottom).

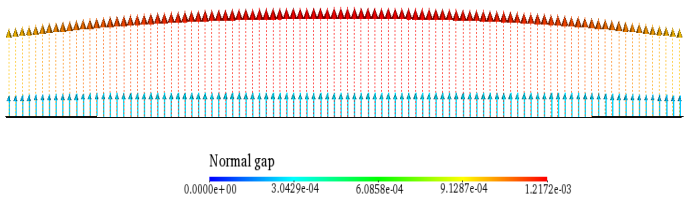


Fig. 7. Comparison of interpenetration at the crack surface obtained using Nitsche's method (shown in blue) and the penalty method with a finite penalty (shown in red).

3.3. *3D Frictionless Contact*

We now extend the 2D plane-stress frictionless contact considered earlier in Section 3.2 and extend it to a three-dimensional setting. The material has a Young's modulus of E = 10 Pa and a Poisson's ratio of ν = 0.3. The crack surface is the plane z = 0.5 and is assumed to be frictionless. We clamp both the top and bottom surfaces and apply a displacement of u_z = -0.1 m on the top surface.

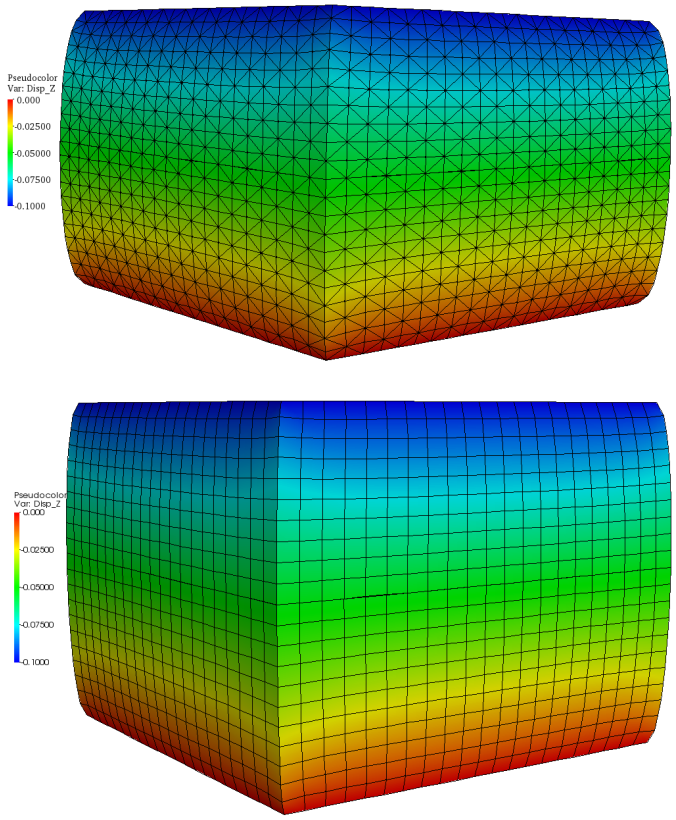


Fig. 8. z-displacement contours for the 3D horizontal crack under frictionless sliding using Nitsche's method with tetrahedral (top) and hexahedral elements (bottom)

We plot the z-displacement contours for Nitsche's method using both hexahedral and tetrahedral elements in Figure 8. In Figure 9, we also plot the normal contact pressure obtained using both Nitsche's method and penalty method with a penalty parameter of $\alpha_N = 10^{16}$ Pa/m for tetrahedral elements. The checkerboarding pattern in contact pressures is evident for large penalties while Nitsche's method returns smooth pressures. Similar results were obtained with hexahedral elements as well though we have omitted them here to avoid repetition.

3.4. *Compressive Fracture with Frictionless Sliding on Crack Faces*

As a final example, we demonstrate the performance of the method in modeling compressive fracture under frictionless sliding. We follow Nemat-Nasser and Horii [21] and consider a single inclined flaw in a square elastic block. The diagonal crack extends from (x,y) = (0.3 m, 0.3 m) to (x,y) = (0.7 m, 0.7 m). The top and bottom surfaces are clamped laterally while the bottom surface is clamped vertically as well. The loading is applied by compressing the top surface by applying a uniform displacement of u_y = - 0.1 m. Further, the

Young's modulus and Poisson's ratio as chosen as $E = 1$ GPa and $\nu = 0.3$ respectively. We consider linear elastic fracture mechanics (LEFM) and employed the critical stress intensity factor approach to advance the crack tip. The propagation direction was determined using a maximum circumferential stress criterion for mixed-mode fracturing (see Fu *et al.* [22] for details).

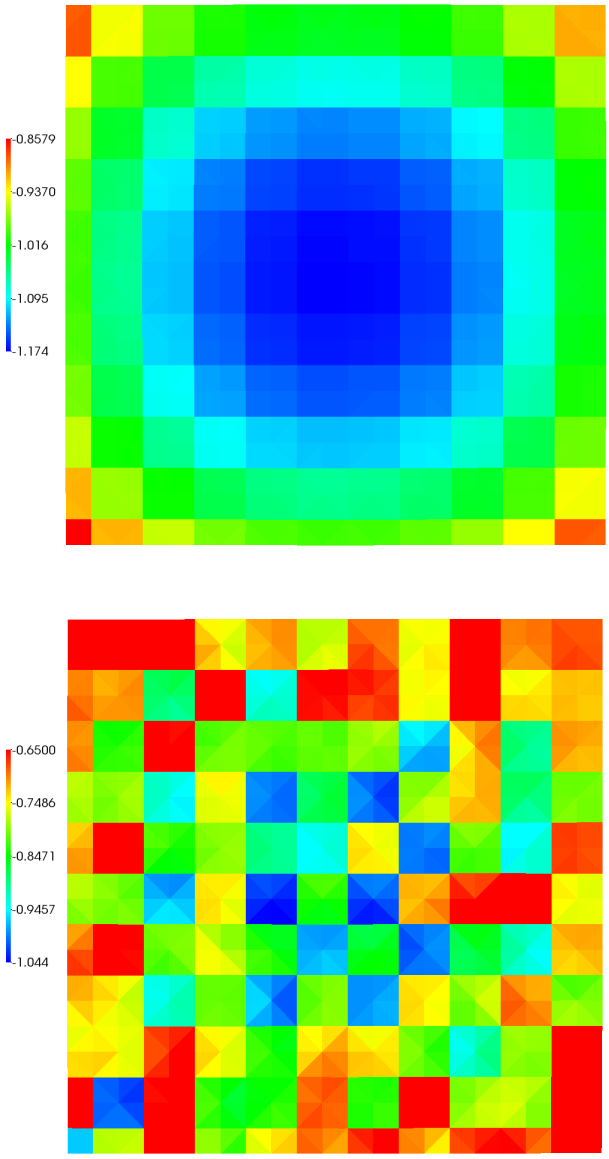


Fig. 9. Normal contact pressure distribution on the crack surface obtained using Nitsche's method (top) and Penalty method with a large penalty (bottom)

In Figure 10, we compare the results obtained using our simulation with the experimental results of Nemat-Nasser and Horii [21]. As seen from the Figure, new tensile fractures develop at the tip of the pre-existing flaw and develop ``winged-fractures" that curve and orient themselves in a direction parallel to the direction of applied loading.

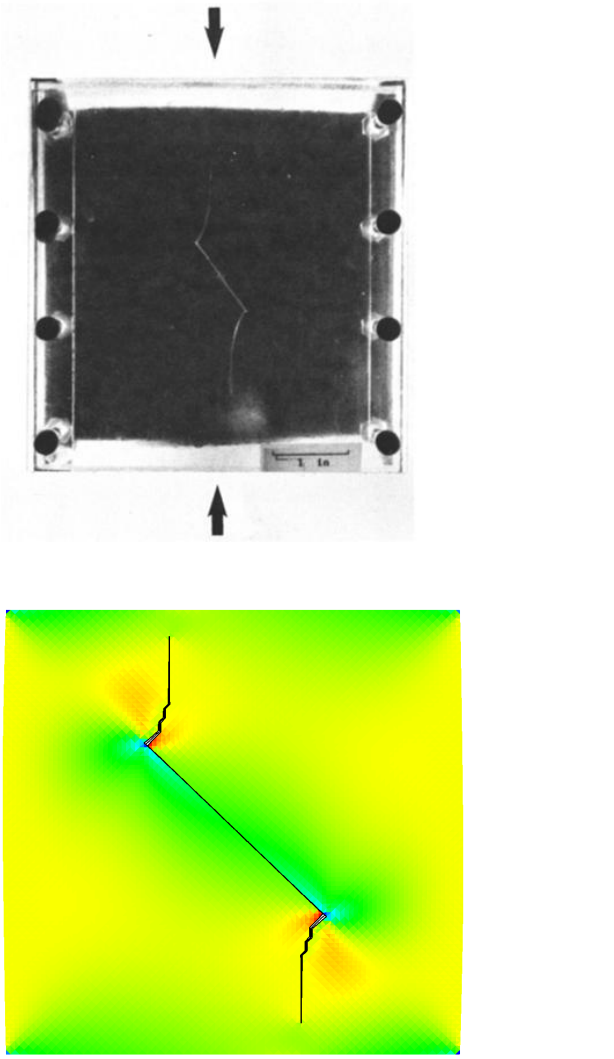


Fig. 10. Crack trajectory for a compressively loaded pre-existing diagonal fracture. The left pane shows the experimental results of Nemat-Nasser and Horii [21] and the right pane shows the results of our numerical model.

Finally, in Figure 11 we consider multiple flaws oriented at 45 degrees to the horizontal axis and load them compressively and examine the overall failure pattern. We notice that the overall failure mechanism remains similar to when the specimen had a single flaw with tensile cracks developing at the tips of pre-existing flaws and curving towards the direction of applied compression. However, we now clearly see the effect of pre-existing flaws on the crack trajectory.

4. CONCLUSION

We have proposed a stabilized algorithm based on Nitsche's method to enforce contact constraints over continuous and discontinuous surfaces in the finite element method. The proposed approach offers many computational advantages over the traditional methods.

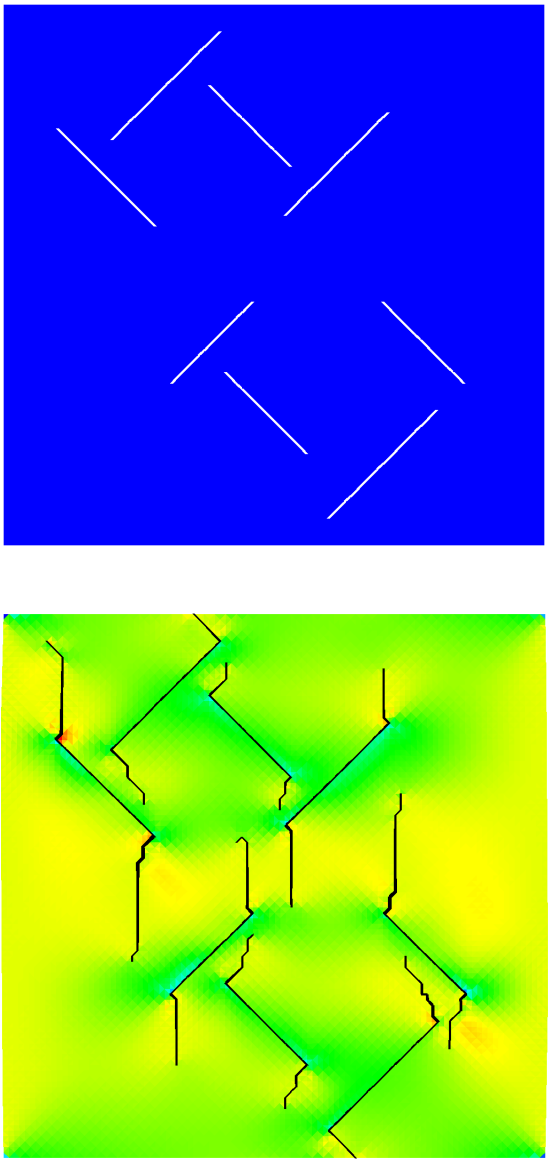


Fig. 11. Failure pattern for a rock specimen under compression in the presence of multiple flaws. The left pane shows the initial configuration and the pre-existing flaws while the right pane shows the crack patterns under compression.

In particular, the proposed method is purely displacement based resulting in a smaller size for the system matrix and eliminates outer augmentation loop required by augmented Lagrange multiplier approaches. The variational consistency of the method and the proposed choice of the stabilization parameter results in well-conditioned system matrices. This allows the proposed method to achieve the correct balance of enforcing the non-interpenetration constraint and recovering contact stresses: a balance often hard to achieve by traditional methods.

Further investigations will consider extending the method to model dissipation at the interface through a Coulombian frictional model in shear. Additionally, extending the method to model Barton-Bandis type behavior in the normal direction could also be of interest to incorporate observed rock joint behavior. Compared with the penalty method, the considerably smaller values used as stabilization parameters could also offer significant computational advantages in explicit calculations by enhancing the CFL time step and hence investigating the method's performance in dynamic fracture is also of interest.

5. ACKNOWLEDGEMENTS

This document was prepared by LLNL under Contract DE-AC52-07NA27344. The authors gratefully acknowledge this support. This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

REFERENCES

1. Ling, H.I., A.H.-D. Cheng. 1997. Rock sliding induced by seismic force. *Int. J. Rock Mech. Min Sci.* 34(6):1021-1029.
2. Bahaaddini, M., G. Sharrock, B.K. Hebblewhite. 2013. Numerical direct shear tests to model the shear behavior of rock joints. *Comput. Geotech.* 51:101-115.
3. Seo, Y.S., G.C. Jeong, J.S. Kim, Y. Ichikawa. 2002. Microscopic observation and contact stress analysis of granite under compression. *Eng. Geology*. 63(3-4):259-275.
4. Scholtès, L., F. Donzé. 2012. Modeling progressive failure in fractured rock masses using a 3D discrete element method. *Int. J. Rock Mech. Min Sci.* 52:18-30.
5. Gehle, C., H.K. Kutter. 2003. Breakage and shear behavior of intermittent rock joints. *Int. J. Rock Mech. Min Sci.* 40(5):687-700.
6. Min, K., L. Jing. 2003. Numerical determination of equivalent elastic compliance tensor for fractured rock masses using the distinct element method. *Int. J. Rock Mech. Min Sci.* 40(6): 795-816.

7. Simo, J., J. Oliver and F. Armero. 1993. An analysis of strong discontinuities induced by strain softening in rate independent inelastic solids. *Comput. Mech.* 12:277-296.

8. Xu, P., A. Needleman. 1994. Numerical simulations of fast crack growth in brittle solids. *J. Mech. Phy. Solids* 42(9): 1397-1434.

9. Camacho, G.T., M. Ortiz. 1996. Computational modelling of impact damage in brittle materials. *Int. J. Solids Struct.* 33(20-22): 2899-2938.

10. Radovitzky, R., A. Seagreaves, M. Tupek, L. Noels. 2010. A scalable 3D fracture and fragmentation algorithm based on a hybrid, discontinuous Galerkin, cohesive element method. *Comput. Meth. Appl. Mech. Eng.* 200(1-4): 326-344.

11. Möes, N., J. Dolbow, T. Belytschko. 1999. A finite element method for crack growth without remeshing. *Int J. Num. Meth. Eng.* 46(1):131-150.

12. Liu, F., R.I. Borja. 2010. Stabilized low-order finite elements for frictional contact with the extended finite element method. *Comput. Meth. Appl. Mech. Eng.* 199(30-40):2456-2471.

13. Dolbow, J., N. Möes, T. Belytschko. 2001. An extended finite element method for modeling crack growth with frictional contact. *Comput. Meth. Appl. Mech. Eng.* 190(51-52):6825-6846.

14. Wriggers, P. 2002. *Computational Contact Mechanics*. 2nd Ed. New York: Springer.

15. Laursen, T. A. 2002. *Computational Contact and Impact Mechanics*. 1st Ed. New York: Springer.

16. Nitsche, J. 1971. Über ein Variationsprinzip zur Lösung von Dirichlet--Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. Abh. Math. Univ. Hamburg. 36: 9-15.

17. Annavarapu C., M. Hautefeuille, J.E. Dolbow. 2014. A Nitsche stabilized finite element method for frictional sliding on embedded interfaces. Part I: Single interface. *Comput. Meth. Appl. Mech. Eng.* 268(1): 417-436.

18. Annavarapu C., M. Hautefeuille, J.E. Dolbow. 2013. A Nitsche stabilized finite element method for frictional sliding on embedded interfaces. Part II: Intersecting interfaces. *Comput. Meth. Appl. Mech. Eng.* 267(1): 318-341.

19. Annavarapu C., M. Hautefeuille, J.E. Dolbow. 2012. A robust Nitsche's formulation for interface problems. *Comput. Meth. Appl. Mech. Eng.* 225-228: 44-54.

20. Papadopoulos, P., R.L. Taylor. 1992. A mixed formulation for the finite element solution of contact problems. *Comput. Meth. Appl. Mech. Eng.* 94(3): 373-389.

21. Nemat Nasser, S., H. Horii. 1982. Compression induced non-planar crack extension with application to splitting, exfoliation and rockburst. *J. Geophy. Res.* 87(B8): 6805-6821.

22. Fu, P., S.M. Johnson, C.R. Carrigan. 2012. An explicitly coupled hydro-geomechanical model for simulating hydraulic fracturing in arbitrary discrete fracture networks. *Int. J. Numer. Anal. Meth. Geomech.* 37(14): 2278-2300.

Computers & Geosciences 58 (2013) 69–79

Contents lists available at SciVerse ScienceDirect

Computers & Geosciences

journal homepage: www.elsevier.com/locate/cageo

Surrogate-based optimization of hydraulic fracturing in pre-existing fracture networks

Mingjie Chen [a,*], Yunwei Sun [a], Pengcheng Fu [a], Charles R. Carrigan [a], Zhiming Lu [b], Charles H. Tong [c], Thomas A. Buscheck [a]

[a] Atmospheric, Earth and Energy Division, Lawrence Livermore National Laboratory, P.O. Box 808, L-223, Livermore, CA 94551, USA
[b] Earth and Environmental Sciences Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA
[c] Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94551, USA

ARTICLE INFO

Article history:
Received 21 March 2013
Received in revised form
16 May 2013
Accepted 17 May 2013
Available online 28 May 2013

Keywords:
Hydraulic fracturing
Fractal dimension
Surrogate model
Optimization
Global sensitivity

ABSTRACT

Hydraulic fracturing has been used widely to stimulate production of oil, natural gas, and geothermal energy in formations with low natural permeability. Numerical optimization of fracture stimulation often requires a large number of evaluations of objective functions and constraints from forward hydraulic fracturing models, which are computationally expensive and even prohibitive in some situations. Moreover, there are a variety of uncertainties associated with the pre-existing fracture distributions and rock mechanical properties, which affect the optimized decisions for hydraulic fracturing. In this study, a surrogate-based approach is developed for efficient optimization of hydraulic fracturing well design in the presence of natural-system uncertainties. The fractal dimension is derived from the simulated fracturing network as the objective for maximizing energy recovery sweep efficiency. The surrogate model, which is constructed using training data from high-fidelity fracturing models for mapping the relationship between uncertain input parameters and the fractal dimension, provides fast approximation of the objective functions and constraints. A suite of surrogate models constructed using different fitting methods is evaluated and validated for fast predictions. Global sensitivity analysis is conducted to gain insights into the impact of the input variables on the output of interest, and further used for parameter screening. The high efficiency of the surrogate-based approach is demonstrated for three optimization scenarios with different and uncertain ambient conditions. Our results suggest the critical importance of considering uncertain pre-existing fracture networks in optimization studies of hydraulic fracturing.

1. Introduction

Hydraulic communication is a key factor for determining hydrocarbon or thermal energy recovery sweep efficiency in an underground reservoir. Sweep efficiency is a measure of the effectiveness of heat, gas or oil recovery process that depends on the volume of the reservoir contacted by an injected fluid. In the petroleum industry, hydraulic fracturing techniques have been used for over 60 years to increase hydraulic communication and stimulate oil and gas production (Britt, 2012). Artificial (stimulated) hydraulic fractures are usually initiated by injecting fluids into the borehole to increase the pressure to the point where the minimal principal stress in the rock becomes tensile. Continued pumping at an elevated pressure causes tensile failure in the rock, forcing it to split and generate a fracture that grows in the direction normal to the least principal stress in the formation. Hydraulic fracturing activities often involve injection of a fracturing fluid with proppants in order to better propagate fractures and to keep them open (Britt, 2012). The design of fracturing treatment should involve the optimization of operational parameters, such as the viscosity of the fracturing fluid, injection rate and duration, proppant concentration, etc., so as to create a fracture geometry that favors increased sweep efficiency. The net present value (NPV) introduced by Ralph and Veatch (1986) as the economic criteria, is usually used as an objective for optimal fracturing treatment design. Some studies have been reported to use a sensitivity-based optimization procedure coupled with a fracture propagation model and an economic model to optimize design parameters leading to maximum NPV (Balen et al., 1988; Hareland et al., 1993; Aggour and Economides, 1998). Nevertheless, this procedure, requiring brute-force parameter-sensitivity analysis, is tedious and incapable of exploring parameter space globally, which could potentially lead to the problem of converging to a local minimum of the objective function.

Rueda et al. (1994) optimized fracturing variables, including the injected fluid volume, injection rate, fluid and proppant type, by

* Corresponding author. Tel.: +1 925 423 5004; fax: +1 925 423 0153.
E-mail addresses: cmj1014@yahoo.com, chen70@llnl.gov (M. Chen).

0098-3004/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cageo.2013.05.006

applying a mixed integer linear programming (MILP) approach, which also lacks a global optimization capability. Mohaghegh et al. (1999) proposed a surrogate-based optimization approach by using a genetic algorithm to fit the dataset generated from a fracturing simulator that models both fracture propagation and proppant transport. Surrogate-based optimization refers to the idea of speeding optimization processes by using fast surrogate models. Surrogate-based optimization approaches have been extensively studied in the past decade for applications in various fields (e.g., Queipo et al., 2005; Wang and Shan, 2007; Forrester and Keane, 2009). Ensemble surrogate methods are also actively studied to achieve more robust approximation by surrogate models (Goel et al., 2007; Sanchez et al., 2008). Queipo et al. (2002) applied a neural network algorithm to construct a "surrogate" of the NPV for an optimal design of hydraulic fracturing treatments. The objective function (NPV) was trained as a function of inputs by a synthetic dataset produced from a high-fidelity physics model, which integrated a fracturing simulator, a proppant transport and sedimentation model, a post-fracturing production model, and an economic model. This surrogate-based procedure is computationally less expensive for obtaining global minimum without executing physics-model simulations, which are computationally prohibitive in some optimizations. However, none of these studies has considered optimizing the hydraulic fracturing of a pre-existing fracture network, which is a very common feature of rocks (Odling, 1992). Moreover, uncertainties of geomechanical properties and of the pre-existing fracture networks, resulting from the geologic architecture and fracture properties, such as fracture density, length, and orientation, etc. (Reeves et al., 2008), have not been rigorously studied for the optimization of hydraulic fracturing treatment.

It has been demonstrated from field studies that fluid flow in fractured rock is primarily controlled by the fracture geometry and the interconnectivity between fractures (Long and Witherspoon, 1985; Cacas et al., 1990). A fractal is a self-similar geometric set (Mandelbrot, 1982) with Hausdorff–Besicovitch dimension exceeding the topological or Euclidian dimension, which is called fractal dimension. It is well recognized that natural fracture networks are fractal over a wide scale range (Barton, 1995; Bonnet et al., 2001), and fractal dimensions have been demonstrated to be efficient metrics for natural fracture patterns (e.g., LaPointe, 1988; Barton, 1995; Berkowitz and Hadad, 1997).

In this work, a surrogate-based optimization approach is proposed for optimizing hydraulic fracturing design in the presence of uncertainties in a pre-existing natural fracture network and its geomechanical properties. A state-of-the-art 2-D hydraulic fracturing code, GEOS-2D (Fu et al., 2012), is used to simulate dynamic fracture propagation within a pre-existing facture network. Instead of integrating physical models and economic models to maximize NPV as the objective function, we focus on physical criteria, that is, the optimal hydraulic fracture propagation under uncertain natural conditions. The fractal dimension of the connected fractures can be derived from the post-fracturing network simulated by GEOS-2D to represent the network density and connectivity. More importantly, the scale-invariant feature of fractals allows observations from the core scale to be applied in another scale (e.g., reservoir scale). Therefore, the fractal dimension is chosen as the objective function to optimize the hydraulic fracturing well design. While a line, square, and cubic have the integer dimensions of 1, 2, and 3, respectively, the fractals in this study, which are applied to linear fractures in a 2-D plane, have a non-integer fractional dimension between 1 and 2.

In this paper, both non-parametric and parametric algorithms are used to construct surrogate models. Both types of surrogate models are quantitatively evaluated for prediction performance by cross-validations, and the best quality model is then selected for optimization. BOBYQA (Powell, 2009), a powerful and efficient derivative-free nonlinear optimization algorithm, is applied to drive a global search on the surrogate-modeled response surface. Compared to previous studies, our optimization methodology includes advances in (1) incorporating uncertain pre-existing natural fracture networks, (2) constructing both non-parametric and parametric surrogate models and conducting rigorous quality evaluations, (3) applying the high-efficient state-of-the-art optimizer, BOBYQA, and (4) deriving the scale-invariant fractal dimension as the objective function.

2. Surrogate-based optimization approach

The proposed surrogate-based approach includes the following key steps (Fig. 1).

1. Populate sample points in parametric space.
2. Setup numerical models and run simulations on those sample points generated in the previous step.
3. Calculate the objective function from the simulated results.
4. Construct and validate surrogate models using the data from the previous steps for predication.
5. Perform optimization using selected surrogate model.

2.1. Sampling in parameter space

As shown in Fig. 1 and Table 1, an 11-dimensional parameter space is constrained by the ranges of the 11 input parameters. Latin Hypercube Sampling (LHS) procedure is used to draw N samples in the designed space following probability distribution functions (PDF) for each parameter. LHS is an effective stratified sampling approach in a high-dimensional space ensuring that all portions of a given partition are sampled (McKay et al., 1979). Each point in the parameter space represents a deterministic vector for the 11 input variables. Fig. 1 shows an example of a 3-D parametric space, in which $N=800$ sample points are generated from the uniform distribution within specified parameter ranges.

2.2. Hydraulic fracturing simulations

In this step, the computationally expensive physical models are constructed and executed N times with each input configuration sampled in the previous step. On each sample point, an initial fracture network is generated and the corresponding hydraulic fracturing is simulated. The initial discrete fracture network is generated with fracture lengths controlled by the Pareto distribution (Odling, 1997)

$$P(L>l)=C\cdot l^{-a} \tag{1}$$

where P is the probability of a fracture of length larger than l, C is a constant that depends on the minimum fracture length in the system, which is assumed to be 5% of the domain size (100 m) in this study, and a is the power law exponent varying between 1 and 3 for natural fracture networks (Davy, 1993; Renshaw, 1999; Reeves et al., 2008). Typically, the mean fracture length of the fracture network increases as a decreases. Natural fracture networks usually consist of two fracture sets with most fractures in a set oriented in the same direction (LaPointe and Hudson, 1985; Ehlen, 2000). In this study, the fracture orientation refers to the angle between the fracture and the maximum principal stress direction (east). The orientation of the first fracture set ranges between 0° and 135°, while that of the second set is always 45° more than the first one. For example, the orientation of the first fracture set in the pre-existing fracture network shown in Fig. 1 is

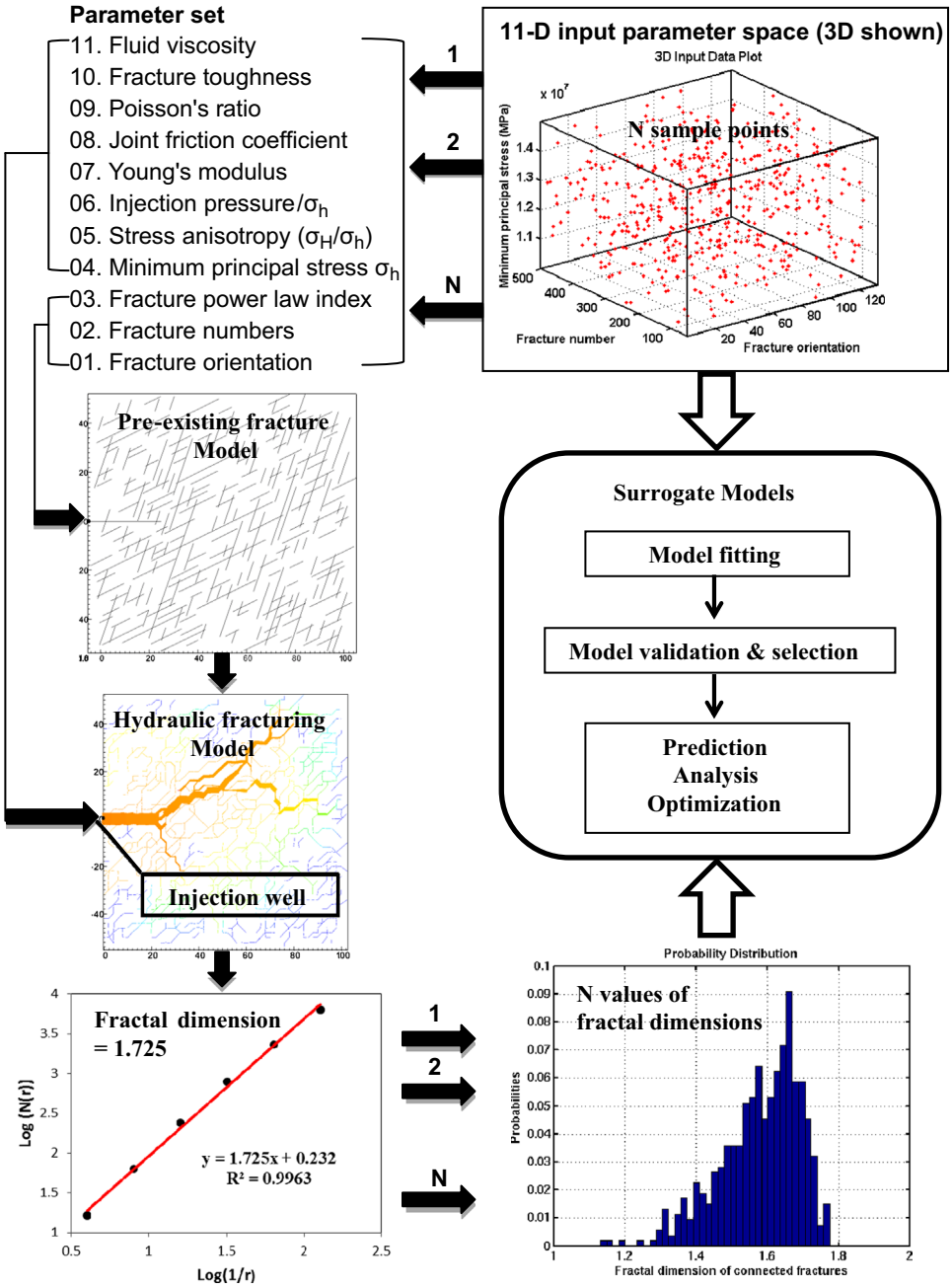


Fig. 1. Surrogate-based modeling approach for simulated hydraulic fracturing.

Table 1
Preliminary experiment: parameter importance ranking for the fractal dimensions of opened fractures in post-fracking networks according to Sobol' total sensitivity indices.

Parameter name	PDF[a]	Min	Max	Sample#1	Indices	Rank
11. Fluid viscosity (Pa s)	Log–U	0.0001	0.001	0.00025	0.51	1
6. Injection pressure/σ_h	U	1	2	1.7	0.43	2
1. Fracture orientation (deg)	U	0	135	25	0.050	3
2. Initial fracture numbers	U	50	500	250	0.031	4
7. Young's modulus (GPa)	U	5	50	31	0.026	5
4. Minimum principal stress σ_h (MPa)	U	10	15	10.1	0.022	6
5. Stress anisotropy (σ_H/σ_h)	U	1	2	1.3	0.014	7
9. Poisson's ratio	U	0.1	0.5	0.2	0.0024	8
3. Fracture power law exponent	U	1	3	1.8	0.001	9
8. Joint friction coefficient	U	0.5	1.2	0.7	0.0	10
10. Fracture toughness (MPa $m^{0.5}$)	U	0.2	2.0	1.0	0.0	11

[a] U and Log-U denote uniform and log-uniform distribution.

25° from the input sample, hence that of the second set is 70°, with 45° from the first set.

Hydraulic fracturing under injected fluid pressure is simulated using an explicitly coupled hydro-geomechanical code, GEOS-2D, developed at Lawrence Livermore National Laboratory (Fu et al., 2012). This code couples a solid solver, a flow solver, a joint module, and a re-meshing module, and is capable of dynamically simulating fracture propagation in a pre-existing fracture network. Fig. 1 presents the simulated fracture distribution after hydraulic fracturing with an injection well located at (0, 0), at sample point 1 with parameter values provided in Table 1.

2.3. Fractal dimension calculation

The fractal dimension of fractures opened by pressurized fluids can be reasonably representative of the density and connectivity of the network. Owing to self-similarity of fractals, the fractal dimension calculated from borehole samples can be extrapolated to reservoir-scale fracture networks. Due to these attractive features, the fractal dimension calculated from the simulated post-fracturing distribution is used as objective function of surrogate models for optimization. The box-counting method is used to measure the fractal dimension of the fracture network (Barton and Larsen, 1985; Chilès, 1988; Walsh and Watterson, 1993). It involves overlaying the fracture network with a sequence of grids with varying cell size r, and counting the number of occupied cells $N(r)$. The number of cells of side length r needed to cover the fracture network is approximated as a power law relation

$$N(r) = k \cdot (1/r)^D, \tag{2}$$

where k is a constant and D is the fractal dimension. By log-transforming the both sides, we obtain

$$\mathrm{Log}(N(r)) = D \cdot \mathrm{Log}(1/r) + \mathrm{Log}(k). \tag{3}$$

Thus, the fractal dimension D can be derived as the slope of the line linearly regressed from a series of size r and the corresponding $N(r)$. Fig. 1 shows that the fractal dimension of the simulated network is 1.725 from the well-fitted regression line with an R^2 value of 0.9963.

2.4. Surrogate-based optimization

Since surrogate models can be quickly constructed once the expensive training dataset is generated, we build alternatives from which the best one is selected according to the model validation results. The selected surrogate model is then used for evaluating objective functions for optimization or for other analyses.

2.4.1. Surrogate model construction

The calculated fractal dimensions, paired with the corresponding sample inputs, constitute the training data set for construction of the non-linear relations between them. For n paired observations, the model is given by

$$Y_i = f(x_i) + \varepsilon_i, \quad i = 1 \text{ to } N. \tag{4}$$

Here, x_i is the input variable vector of sample i, Y_i is the response observation (calculated fractal dimension), $f(x_i)$ is the mean response, ε_i is the error, and N is the sample number. Generally speaking, there are two kinds of fitting methods, namely, parametric and non-parametric regression. The parametric approaches, such as Gaussian Process (GSP) and Polynomial Regression (PRG), presume a uniform global function form between input variables and the response variable, and require the estimation of a finite number of coefficients (Williams and Rasmussen, 1996; Draper and Smith, 1998), while non-parametric approaches, such as Multivariate Adaptive Regression Splines (MARS), use different types of local models in different regions of the data to construct the overall model (Friedman, 1991). In our approach, we build MARS, GSP, and PRG models and determine which one performs the best by follow-up validation. Various PRG models are also built with different order and different number of input variables that are the most sensitive ones ranked by global sensitivity analysis to be discussed in the next section. The first, second, and third order PRG including N_v input variables can be expressed as

$$\begin{aligned} f_1(\mathbf{x}) &= \beta_0 + \sum_{i=1}^{N_v} \beta_i x_i, \\ f_2(\mathbf{x}) &= f_1(\mathbf{x}) + \sum_{i=1}^{N_v} \sum_{j=i}^{N_v} \beta_{ij} x_i x_j, \\ f_3(\mathbf{x}) &= f_2(\mathbf{x}) + \sum_{i=1}^{N_v} \sum_{j=i}^{N_v} \sum_{k=j}^{N_v} \beta_{ijk} x_i x_j x_k, \end{aligned} \tag{5}$$

where β_0, β_{ij}, β_{ijk} are coefficients to be estimated. Higher order PRG can be formulated by adding higher-order terms. With more input variables included in higher order PRG, the fitting is better, but the number of coefficients increases, which must be less than the number (N) of observations (training dataset). Because of the limited training data, there is a trade-off between the order of PRG and the number of included variables for the best fit.

2.4.2. Global sensitivity analysis

Sensitivity is a measure of the contribution of an independent variable to the total variances of the dependent variable. Sensitivity analysis of a model system can be used as the following purposes.

1. Parameter screening: fix one or more of the input variables with negligible influence on the output variability.
2. Variable prioritization: rank input variables according to their sensitivity indices.
3. Variable selection for reducing uncertainty: invest money to measure those sensitive variables that can reduce output uncertainty to maximum extent.

There are numerous methods for sensitivity analysis (Frey and Patil, 2002), among which the Sobol' (1993) method is used to drive global sensitivity analysis of input variables for the output variable, i.e., the fractal dimension. Sobol' method is a variance-based sensitivity analysis, which decomposes the variances of the output into fractions attributed to each input (first-order indices) and their interactions (second- or higher-order indices). These fractions are interpreted as the sensitivities. Sobol' total sensitivity measures the contribution to the output variances of each input variable, including all variances caused by its interactions with any other input variables in all the orders. Using the training dataset, Sobol's total sensitivity indices can be calculated to measure the relative importance of each input variable to the output of the hydraulic fracturing system. In this study, the sensitivity analysis for the preliminary experiment screens out the non-sensitive parameters to reduce the parameter dimension for the 2nd stage experiment of optimization. The selection of input variables from the reduced-dimension parameter space in the PRG models is also based on the parameter ranking by Sobol' indices.

2.4.3. Model validation and selection

A well-fitted surrogate model does not necessarily mean that it is good for prediction. It is easy to over-fit data by including too many degrees of freedom. One way to measure the predictive ability of a surrogate model is to test it using a test dataset, which is split from the sample data and not used in training. Nevertheless, it will limit

the data available for constructing the surrogate models. Alternatively, the popular leave-one-out cross-validation (LOOCV) method can make use of the available sample data much more efficiently (Picard and Cook, 1984). Given N input samples, a surrogate model is constructed N times efficiently, each time leaving out one of the input sample from training, and using the omitted sample to test the model. The generalization error of the LOOCV can be estimated using the root mean square error (RMSE)

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Y_i - f_i^{(-i)})^2}{N}} \quad (6)$$

where Y_i represents the ith response observation (calculated fractal dimension), and $f_i^{(-i)}$ denotes the prediction (interpolated fractal dimension) tested by sample i using the surrogate model fitted by all the other N-1 samples. The surrogate model with a minimum RMSE is selected for optimization.

2.4.4. Optimizer

Bound Optimization BY Quadratic Approximation (BOBYQA) algorithm is applied to search the minimal objective function (negative fractal dimension) of the surrogate model $f(\mathbf{x}), \mathbf{x} \in \mathcal{R}^N$, where $\mathcal{R}^N$ is the N-dimensional parameter space constrained by the range of each input variable. BOBYQA is a powerful numerical optimization solver for derivative-free nonlinear problems, subject to simple bound constraints (Powell, 2009). In the case studies, optimal hydraulic fracturing design parameters and natural field properties corresponding to the minimal objective function are found on the response surface using BOBYQA optimizer.

2.5. Implementation

The proposed approach was implemented in a Python code that couples the hydraulic fracturing simulator GEOS-2D (Fu et al., 2012) with the uncertainty quantification tools contained within the PSUADE code (Tong, 2009). PSUADE (Problem Solving environment for Uncertainty quantification And Design Exploration) is a suite of uncertainty quantification modules capable of addressing high-dimensional sampling, parameter screening, global sensitivity analysis, response surface analysis, uncertainty assessment, numerical calibration, and optimization (Hsieh, 2007; Wemhoff and Hsieh, 2007; Sun et al., 2012). The computationally expensive hydraulic fracturing simulations for generation of the synthetic training dataset (GEOS-2D) are executed using the high performance computing facilities at Lawrence Livermore National Laboratory (LLNL). The hundreds of runs are distributed to a LLNL cluster equipped with Intel 6-core Xeon X5660 processors, 96 nodes, and RAM with 48 GB/node. The box-counting method for deriving fractal dimension of connected fractures from the post-fracturing distribution is implemented in a Fortran code.

3. Case study: hydraulic fracturing well design optimization

In this section, the developed surrogate-based approach is applied to optimizing the hydraulic fracturing well design (location and length) in a 2-D domain under uncertain natural-system conditions. To reduce the dimensionality of the input parameter space, preliminary simulations are performed to generate a training dataset used to conduct global sensitivity analysis for parameter screening. The input parameter sampling and numerical simulations are presented in Fig. 1 and Table 1. Based on N=800 observation pairs, Sobol' total sensitivity indices are derived and parameter importance is ranked (Table 1). Of the Nv=11 input parameters, two operational ones, working fluid viscosity and injection pressure, are found to be the most important for effective fracturing. The four least sensitive parameters with Sobol' indices less than 0.01 are screened out. The remaining variables—two parameters related to pre-existing network, fracture orientation and number, and three parameters related to rock mechanical properties, Young's modulus, minimum principal stress, and stress anisotropy, are included for the optimization experiment described below.

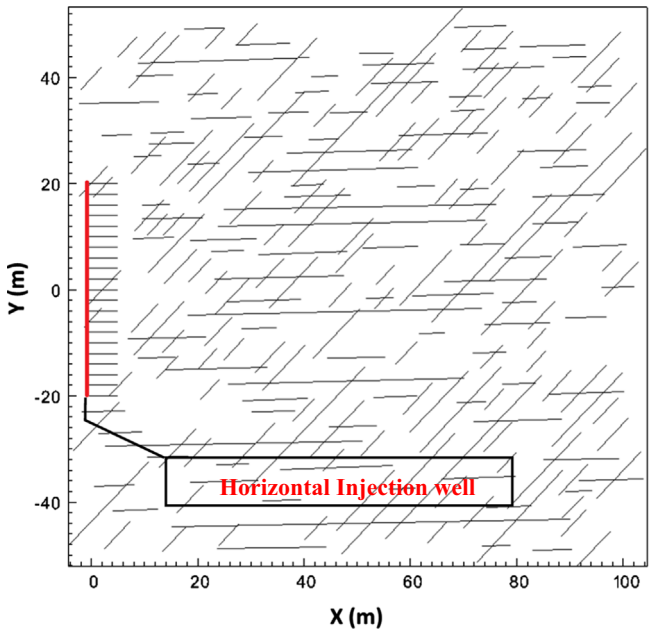


Fig. 2. An example of a horizontal well (center at y=0 m and length=40 m) placed in a pre-existing network (orientation =0° and number =250). The red solid line is horizontal injection well with uncertain location and length along left y-axis. The pre-existing fracture orientation and number of natural network are also uncertain. The maximum and minimum principal stress are assumed x- and y-direction, respectively. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

3.1. Experimental design

As illustrated in Fig. 2, a horizontal injection well is placed in an experimental 2-D physical domain along its left-most boundary (along the y-axis). The pertinent design parameters of interest here include the length of the open (perforated) injection interval (anywhere from 0 to 40 m) and its center lying between y=−20 and 20 m. The design parameters and the five most important natural-system parameters determined above, are treated as uncertain parameters. A total of 529 input samples are drawn from the seven-dimensional parameter space using the LHS sampling method. Two of the seven parameters, fracture orientation and the number of fractures in the pre-existing network, are fed into the pre-existing fracture model and the remaining five are applied to the hydraulic fracturing model. Instead of injection pressure, injection rate is used as the source term of the fracturing model. The total injection rate is fixed at 0.25 m^3/s, and is averaged over the perforated well length, which is subdivided into 2-m long injection nodes. As a result, the injection rate applied on each injection node decreases linearly with increasing horizontal-well length.

3.2. Synthetic dataset analysis

For each of 529 input samples, pre-existing network are generated and GEOS-2D models are executed, and nine snapshots of post-fracturing network distributions are exported in nine sequential time steps from which the fractal dimensions are

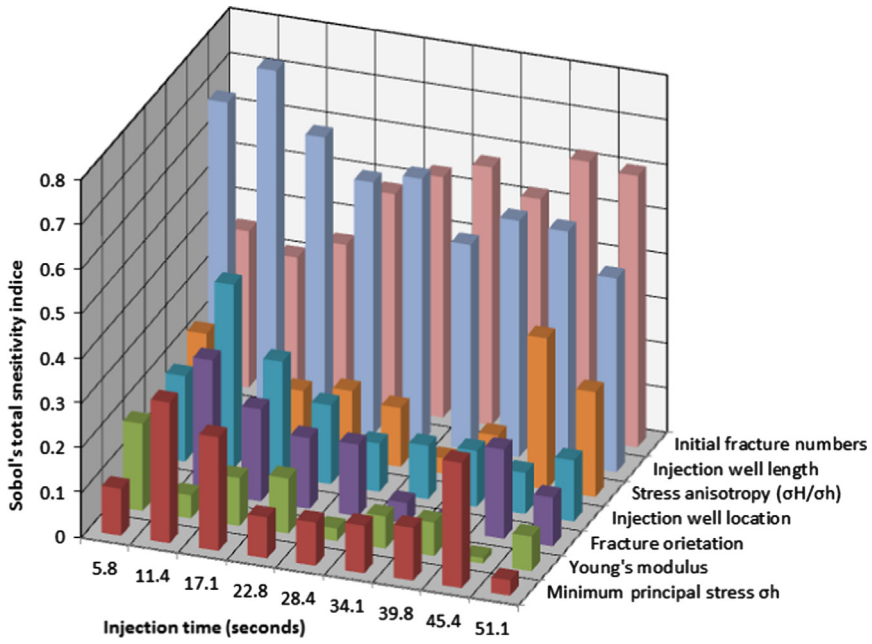


Fig. 3. Global sensitivity of fractal dimension to the 7 input parameters for 9 sequential injection time steps. The 9 parameter sequences are ordered according to the last one.

derived. Mean values of the 529 fractal dimensions increase with the injection time or fluid volume (Fig. 3a), suggesting that fracture networks keep growing with the continuous injection of fluid. The time series of the mean fractal dimensions also indicate that their growth rates are very high initially, and gradually decreases to nearly zero from 11.4 to 51.1 s, suggesting that the economic benefit of hydraulic fracturing declines with time. The probability distribution of the 529 fractal dimension results in the last snapshot at 51.1 s shows that most of them are between 1.5 and 1.7, and the value with highest possibility (10%) is around 1.65 (Fig. 3b). The corresponding cumulative probability indicates that about 25% of 529 fractal dimensions is less than 1.5, 50% less than 1.6, and 75% less than 1.65. Only 10% of these fracture dimension values are above 1.7 and the maximum value is 1.79. The nine sets of 529 observation pairs consisting of the seven input variables and the corresponding fractal dimension are served as the training and testing dataset for surrogate models.

3.3. Global sensitivity analysis

All seven input variables are normalized between zero and one, based on their upper and lower bounds. For each input sample, fracture distributions at nine sequential injection time steps were generated, from which the corresponding fractal dimensions are derived. Fig. 4 shows the global sensitivity of nine sets of fractal dimensions to the seven input variables sorted by the last set. For all the nine time steps, the variability of fractal dimensions is largely influenced by the initial fracture number and well length (Sobol' indices >0.5), and moderately by the other 5 input variables, indicating that the initial fracture number is the key uncertain parameter influencing post-fracturing conditions. Injection lengths (and the corresponding averaged injection rate) are the key contributors to the variability of fractal dimensions at the earlier injection stages, while initial fracture number becomes the key contributor at the later stages. Well center location strongly affects the fractal dimension (Sobol' indices $=0.4$), while becoming marginally important (Sobol' indices $=0.1$) as injection proceeds. Overall, two stress parameters, minimum principal stress, stress anisotropy, and fracture set orientation, influence the objective somewhat more than Young's modulus does. The sensitivity information inferred above is used to rank variable prioritization to be included in PRG models below.

3.4. Surrogate models evaluation

Non-parametric MARS, parametric GSP and 11 PRG models with various parameters and orders are constructed for the nine snapshots, each using 529 observation pairs (input parameters versus fractal dimension). Table 2 shows the comparison of MARS, GSP and 11 PRG models constructed for the post-fracturing distribution (i.e., the last snapshot). The natural-system parameters included in PRG models are determined according to the importance ranking by Sobol' indices (Fig. 3). For examples, minimum principal stress is dropped off for the 6-parameter PRG model, and Young's modulus is further excluded from the 5-parameter PRG model, because the two parameters are ranked as least important for fracturing at the final time step. In terms of fitting error, the more coefficients that are included, the higher the accuracy of PRG models becomes. In fact, when the number of coefficients is greater than 125, PRG models fit the training dataset better than the MARS model does. Nevertheless, the predictive ability, tested against a new dataset, will usually get worse as more terms are included, due to over-fitting. As shown in Table 2, the RMSE of cross-validation for each surrogate model confirms that the best fitted PRG model with 461 coefficients turns out to be the worst in prediction performance, and the quadratic PRG, with seven variables and just 35 estimated coefficients, had the best prediction performance among 11 PRG models. Finally, the MARS model is selected for optimization due to its better prediction performance than both GSP and the best PRG model.

To illustrate the surrogate model quality regarding fitting and validation, the scatter plots of fractal dimension simulated by surrogate models versus GEOS-2D from 529 sample inputs are compared between MARS model and the best-fitted, but worst-validated PRG model (5-order 6-parameter) (Fig. 5). The closer the points are to the diagonal line, the better the surrogate model matches the physical model. It is seen that the points are clustered closely along the diagonal line for the PRG model fitting (RMSE $=0.00805$), but are significantly scattered for cross-validation (RMSE $=0.401$). Conversely, points in both the MARS

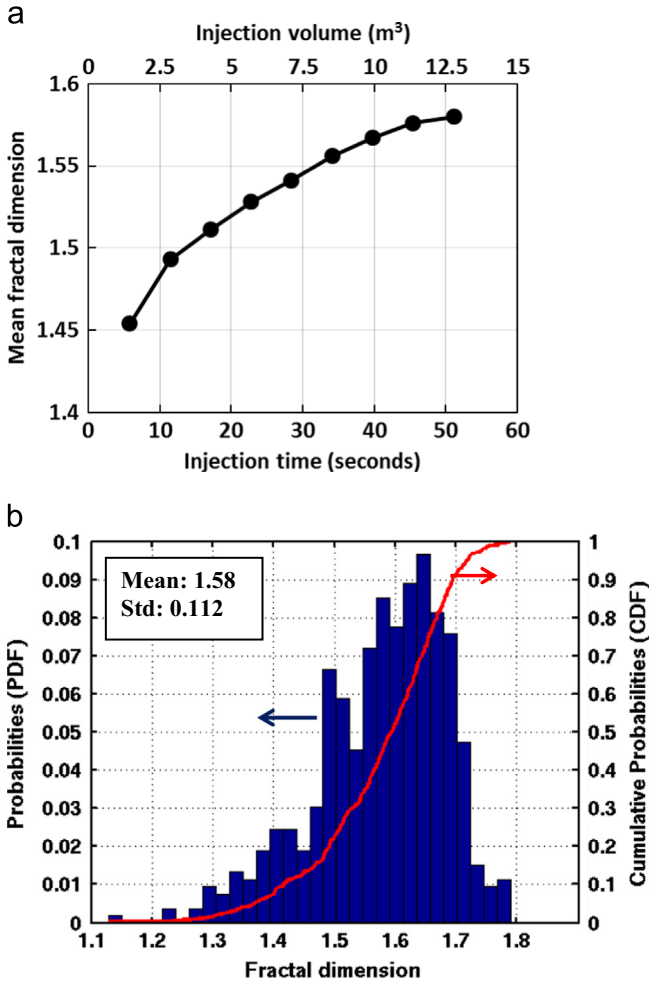


Fig. 4. Statistics of 529 derived fractal dimension: (a) mean for 9 injection time or volumes and (b) PDF and CDF at finial time (51.1 s or 12.8 m^3 injected fluid volume).

Table 2
Evaluation of surrogate models for fracture network at final time.

Construction method	Estimated Coefficients	RMSE	
		Fitting	Validation
MARS	–	0.0257	0.0410
GSP	–	0.0278	0.0428
1-order 7-parameter PRG	7	0.0473	0.0483
2-order 6-parameter PRG	27	0.0390	0.0458
2-order 7-parameter PRG	35	0.0378	0.0436
3-order 5-parameter PRG	55	0.0326	0.0452
3-order 6-parameter PRG	83	0.0300	0.0458
3-order 7-parameter PRG	119	0.0283	0.0462
4-order 5-parameter PRG	125	0.0258	0.0506
4-order 6-parameter PRG	209	0.0221	0.0589
5-order 5-parameter PRG	251	0.0184	0.0568
4-order 7-parameter PRG	329	0.0169	0.0865
5-order 6-parameter PRG	461	0.00805	0.4010

fitting and cross-validation scatter plots are moderately spread with 0.0257 and 0.0410 of RMSE, respectively.

3.5. Horizontal well design optimization

The problem of interest is to find the favorable fracture-stimulation well design variables, namely, well center y location and the perforation length, in the presence of natural-system uncertainty. To investigate how natural-system uncertainty affects optimal well design, three optimization cases with sequentially decreasing natural-system uncertainty are performed for the last snapshot at an injection time of 51.1 s. Case A searches the minimum objective function (maximum fractal dimension) in a 7-D parameter space, with two design variables and with five natural-system variables treated as uncertain. Case B is adapted from case A, with the uncertainty reduced by fixing the fracture orientation and number, which are two parameters describing the pre-existing fracture network. In case C, only well location and length are allowed to vary within the specified ranges during the optimization process, by further fixing the three geomechanical variables affecting fracture propagation, minimal principal stress, stress anisotropy, and Young's modulus. The objective function to be minimized is the negative fractal dimension. All the three optimization cases are efficiently conducted using surrogate models without rerunning the expensive physics-based GEOS-2D, due to the flexibility of our surrogate-based approach. The BOBYQA optimizer, coupled with the selected MARS models, is executed for the three inverse problems.

Fig. 6 depicts the optimization processes, which involves searching the minimal objective function for each of the three cases. It is seen that the number of evaluations of the surrogate model required to satisfy the convergence criteria (10^{-6}) is 337, 269 and 994, respectively. Each of the optimizations requires hundreds of model evaluations and can be completed in less than a minute, while a single realization conducted with the GEOS-2D code costs tens of hours. Moreover, a physics-based model is usually not as smooth as its surrogate, implying that a greater number of model evaluations are required for convergence than required by surrogate-based optimization. As a result, the high-efficient surrogate-based optimization approach can make the otherwise computationally prohibitive procedure practically achievable. An example of an expensive procedure is Bayesian stochastic joint inversion modeling using hard (borehole core) and soft data (geophysical survey), which usually entails expensive Markov Chain Monte Carlo sampling. Another advantage of the surrogate-based approach is its high degree of flexibility. Once the training data is generated from the expensive physics-model simulations, numerous surrogate models can be constructed and validated for optimization within a very short time.

The optimal values of the parameter sets corresponding to the minimum objectives are listed in Table 3. Case A represents a scenario in which the hydraulic fracturing treatment is designed with minimal knowledge of the targeted field; thus, a wide range of the natural-system properties must be accounted for. The optimal location of the well center is found to be 4.31 m on the y-axis, and the optimal well length is 0.08 m. This indicates that, to obtain a maximum fractal dimension, the fluid should be injected in just one injection node at $y=4$ m, and at the rate of 0.25 m^3/s, if fracturing is to be optimized for this level of natural-system uncertainty. With the entire injection rate concentrated at one node, the maximum possible hydraulic pressure is achieved, which confirms our intuition about what will maximize the growth of the fracture network.

Case B assumes that both the fracture orientation and fracture number of the pre-existing network are already determined to be 1° and 250°, respectively, on the basis of borehole core data or other geophysical measurements. The optimal well design parameters (position and length) are found to be 5.09 m and 21.3 m, which corresponds to a hydraulic fracturing scheme where fluid is injected into 11 nodes, centered at $y=5$ m, with each injected at a rate of $0.25/11=0.0227$ m^3/s. Unlike case A, where all of the fluid injection (and pressurization) is concentrated in one node, pressurization in case B is distributed along 11 nodes, suggesting that

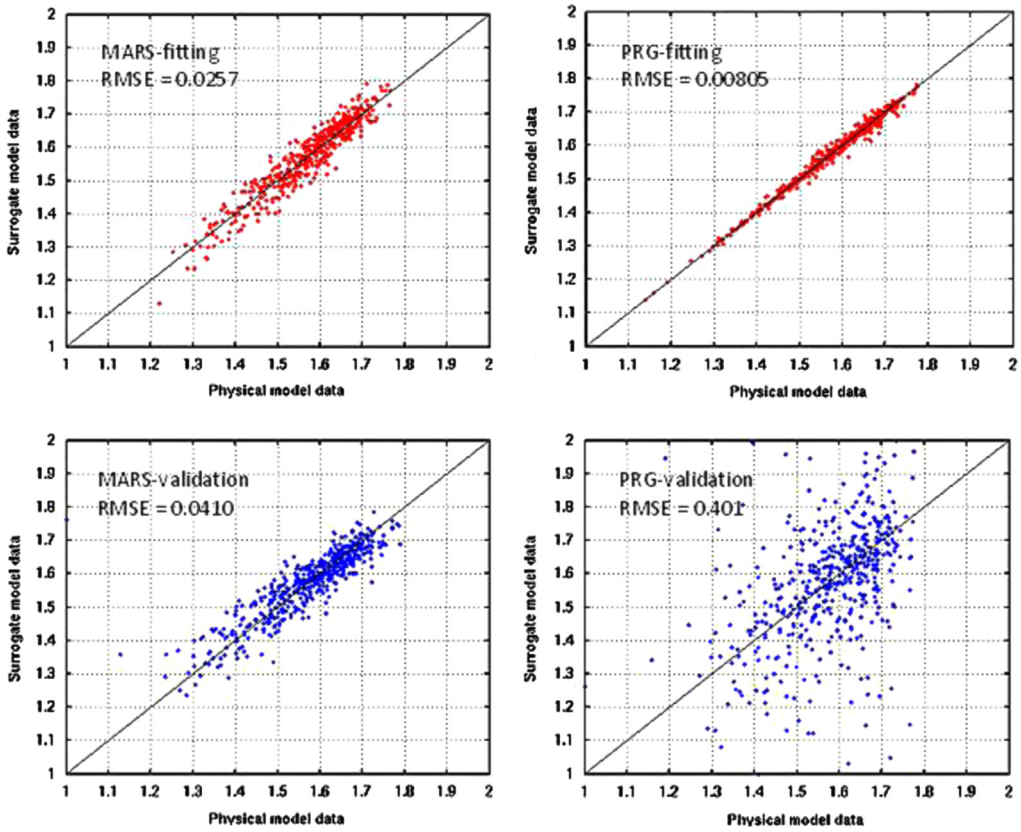


Fig. 5. Scatter plots of fractal dimension simulated using surrogate model data versus physical model from 529 input samples: the comparison of fitting (red dots) and cross-validation (blue dots) between MARS and 5-order 6-parameter PRG surrogate models. The tighter the points clustered along diagonal, the closer the surrogate model data match the physical model data. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

both the distribution and magnitude of pressure are important for creating a favorable fracturing network, and must be traded off given the limited total injection volume. The maximum fractal dimension is 1.622, which has been significantly reduced from 1.872 in case A, demonstrating the importance of considering uncertainty of the pre-existing fracture network for optimizing the hydraulic fracturing treatment. Sensitivity analysis has shown that the fractal dimension is highly sensitive to the initial fracture number (Fig. 3), so it is reasonable to conclude that the large decrease of fractal dimension from case A to B results from the large reduction of the initial fracture number from 486 to 250. It is also seen that fracture orientation and Young's modulus differ a lot from case A to B, but since they were found not to strongly affect fractal dimension, they are not likely to be the main contributors to its decrement.

Case C is designed to investigate the optimal well injection scheme given full knowledge of the natural system, with all five natural-system properties fixed as listed in Table 3. The optimized well injection design parameters turn out to be similar to those in case B, suggesting that uncertainty of the three rock mechanical parameters has a small influence on the optimization results. On the other hand, the comparison with case A shows that the uncertainties of the two input variables for pre-existing fracture network can lead to a big difference in the optimization results. These findings demonstrate the importance of addressing uncertainty of the pre-existing fracture network, rather than addressing that of the rock geomechanical properties in optimizing hydraulic-fracturing treatments, which was lacking in previous studies. The moderate decrement of maximal fractal dimension from case B to C is believed primarily caused by the increment of stress anisotropy from 1.0 to 1.2, based on the fact of its relatively small sensitivity to the other varied rock properties (Fig. 3). The 2-D response surface for case C is shown in Fig. 7. Apparently, multiple local minimal objective functions exist, with the global minimum being found using the BOBYQA optimizer.

Fig. 8 plots the three post-fracturing distributions simulated using the corresponding optimal input parameter sets. It is apparent that the network connected by fluid injection for case A sweeps a larger area than the other 2 cases, demonstrating that the fractal dimension of opened fracture network is an appropriate indicator of the potential energy sweep efficiency in the target field. The fractures in case C propagate mainly along x-axis (maximum principal stress direction) since the stress field is moderately anisotropic while the stresses in case A and B are almost isotropic.

4. Summary and conclusions

A surrogate-based optimization approach involving high-dimensional parameter space sampling, numerical physics-model simulations, objective-function evaluation, surrogate-model construction and validation, with the coupled execution of the optimizer and surrogate models, is developed and implemented for optimizing hydraulic-fracturing decision. For a strongly non-linear process, such as hydraulic fracturing considered in this study, the surrogate model constructed by the non-parametric MARS method is demonstrated to have the best prediction performance according to the cross-validation, and hence was selected for optimizing the hydraulic fracturing treatment. The 3 optimization cases, each requiring hundreds of surrogate model evaluations to meet convergence tolerance, are completed in less than one minute, demonstrating the high efficiency of the approach. A comparison study of 3 optimization cases is conducted by varying the dimensionality of the parameter space

without rerunning expensive physics-model simulations. Moreover, additional optimizations using surrogate models can be performed quickly and easily for particular purposes if necessary, for example, reducing the uncertainty of an input variable by narrowing its range.

The comparison study shows the optimization results which depend on the degree of uncertainty of the pre-existing fracture networks. This indicates the importance of incorporating information about pre-existing fracture networks into the process of optimizing hydraulic fracturing treatment, which has been largely overlooked by previous optimization studies in the literature. In contrast, the influence of uncertainty in rock geomechanical properties on the optimal injection scheme is found to be less important. These findings suggest that the pre-existing fracture network, rather than the geomechanical properties, should be the top priority to be characterized before designing a hydraulic fracturing treatment.

The statistical analysis of the training data and fracture networks for the three optimized hydraulic-fracturing cases indicates that fractal dimension is a useful metric for quantifying the density and connectivity of a fracture network. Furthermore, the scale-invariant nature of the fractal makes it a universal indicator for the fracture network across wide range of spatial scales, from core through outcrop to aerial image scale. The successful incorporation of fractal dimension into the efficient surrogate-based approach in this study provides a useful solution for other inverse problems that suffer from the heavy computational burden and multi-scale measurements, such as the stochastic joint inversion problem.

The decreasing growth rate of the mean fractal dimension with injection time implies the diminishing value of continuing the hydraulic fracturing operation. Therefore, there exists a cost-efficient time to stop the fracturing operation, that is, the injection time and the rate need to be optimized for economic objective. Although this paper is focused on incorporating uncertainty of the natural system into optimization and hence only considers the physical criterion as the objective function, the presented surrogate-based optimization approach, can be modified to find optimal injection rate and time by integrating an energy production model and economic model to derive both physical and economic criteria as the objective function.

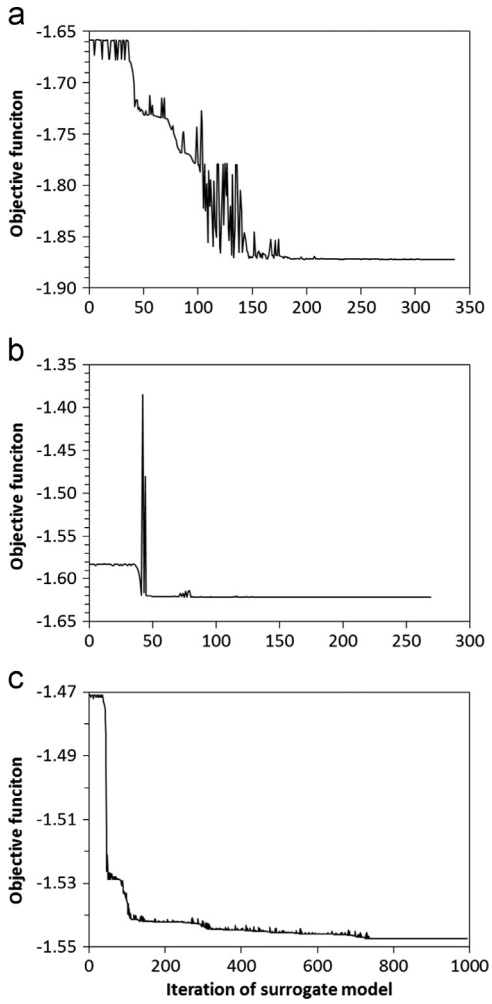


Fig. 6. Minimal objective searching curve for optimization case with (a) 7 uncertain parameters, (b) 5 uncertain parameters, and (c) 2 uncertain parameters. The optimal parameter values for the 3 cases are proved in Table 3.

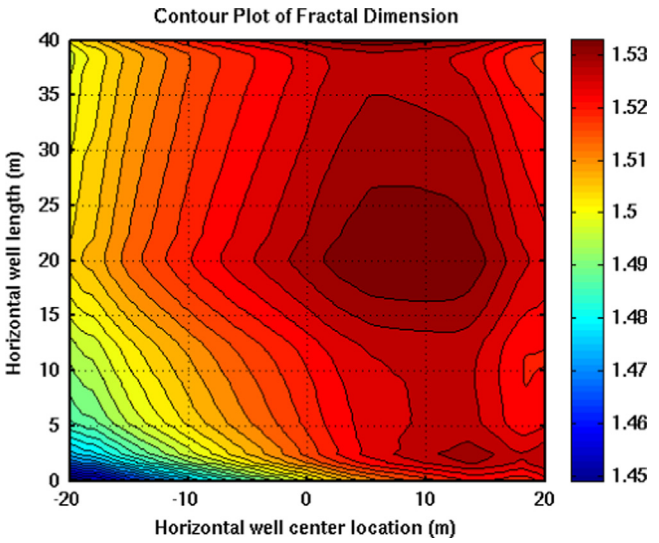


Fig. 7. The visualized response surface with 2 uncertain design parameters, i.e., horizontal well center at *y*-axis and well length.

Table 3
Optimization of well center location and length for fracture network at final time.

Input sample space	Case A: 7-D		Case B: 5-D		Case C: 2-D	
	Range	Opt.	Range	Opt.	Range	Opt.
Fracture orientation	0–135	121	1	–	1	–
Initial fracture number	50–500	486	250	–	250	–
Minimum principal stress σ_h (MPa)	10–15	13.02	10–15	13.27	12.5	–
Stress anisotropy (σ_H/σ_h)	1.0–1.5	1.07	1.0–1.5	1.00	1.2	–
Young's modulus (GPa)	5–50	38.33	5–50	48.65	25	–
Injection well center on *y*-axis (m)	−20 to 20	4.31	−20 to 20	5.09	−20 to 20	5.09
Injection well length (m)	0–40	0.08	0–40	21.3	0–40	21.9
Maximum fractal dimension		1.872		1.622		1.547

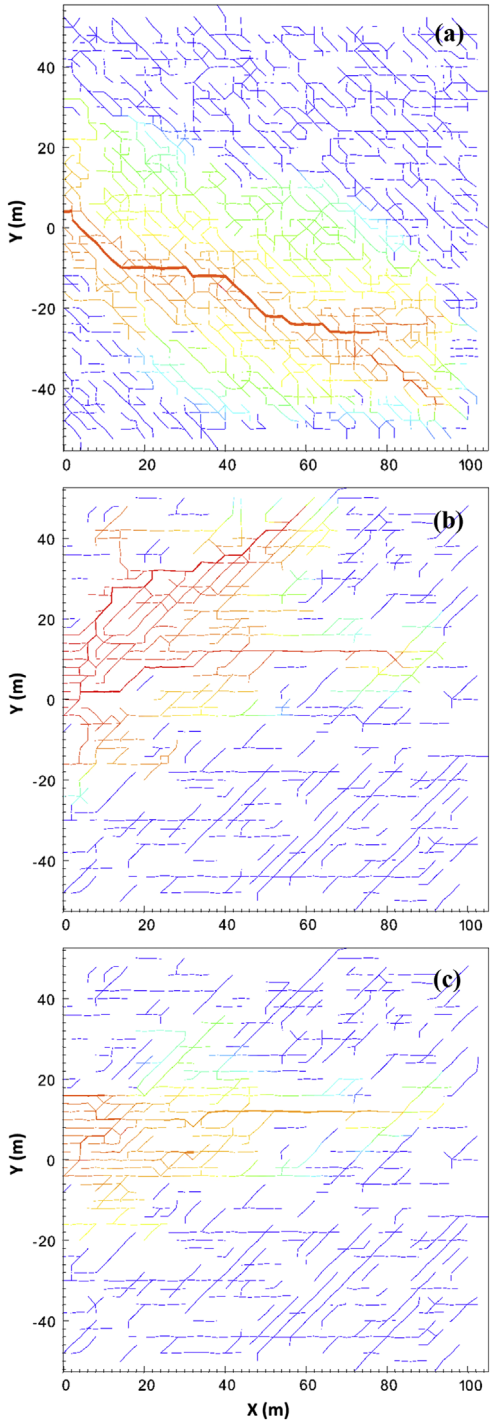


Fig. 8. The post-fracking network corresponding to the optimal parameter set (values provided in Table 3) with (a) 7 uncertain parameters, (b) 5 uncertain parameters, and (c) 2 uncertain parameters. The optimal parameters for the 3 cases are provided in Table 3. The color of the fractures is based on the hydro-pressure. Red indicates the maximum pressure, while blue denotes zero pressure, meaning closed fractures which are not included in fractal dimension calculation. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

Acknowledgments

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory (LLNL) under contract DE-AC52-07NA27344. This work was supported by LDRD-SI program of Lawrence Livermore National Laboratory. We wish to thank Andrew Tompson at LLNL and two anonymous reviewers for their comments that improved the paper.

References

Aggour, T.M., Economides, M.J., 1998. Optimization of the performance of high-permeability fractured wells. In: SPE International Symposium on Formation Damage Control, Lafayette, LA, SPE Paper 39474.

Balen, M.R., Meng, H.Z., Economides, M.J.,1988. Application of net present value (NPV) in the optimization of hydraulic fractures. In: SPE Eastern Regional Meeting, Charleston, South Carolina, USA, SPE Paper 18541, pp. 181–191.

Barton, C.C., Larsen, E., 1985. Fractal geometry of two-dimensional fracture networks at Yucca Mountain, southwestern Nevada. In: Stephannson O. (Ed.), Proceedings of the International Symposium on Fundamentals of Rock Joints, Gentek, Lulea, Sweden, pp. 77–84.

Barton, C.C., 1995. Fractal analysis of scaling and spatial clustering of fractures. In: Barton, P.R., LaPointe, P.R. (Eds.), Fractals in the Earth Sciences. Plenum, New York, pp. 141–178.

Berkowitz, B., Hadad, A., 1997. Fractal and multifractal measures of natural and synthetic fracture networks. Journal of Geophysical Research 103, 12205–12218.

Bonnet, E., Bour, O., Odling, N.E., Davy, P., Main, I., Cowie, P., Berkowitz, B., 2001. Scaling of fracture systems in geological media. Reviews of Geophysics 39 (3), 347–383, http://dx.doi.org/10.1029/1999RG000074.

Britt, L., 2012. Fracture stimulation fundamentals. Journal of Natural Gas Science and Engineering 8, 34–51 http://dx.doi.org/10.1016/j.jngse.2012.06.006.

Cacas, M.C., Ledoux, E., De Marsily, G., Tilie, B., Barbreau, A., Durand, E, Feuga, B., Peaudecerf, P., 1990. Modeling fracture flow with a stochastic discrete fracture network: calibration and validation: 1. The flow model. Water Resources Research 26, 469–489, http://dx.doi.org/10.1029/WR026i003p00479.

Chilès, J.P., 1988. Fractal and geostatistical methods for modeling of a fracture network. Mathematical Geology 20, 631–654.

Davy, P., 1993. On the frequency-length distribution of the San Andreas fault system. Journal of Geophysical Research 98 (B7), 12141–12151.

Draper, N.R., Smith, H., 1998. Applied Regression Analysis, 3rd ed. Wiley Interscience, New York, NY p. 736.

Ehlen, J., 2000. Fractal analysis of joint patterns in granite. International Journal of Rock Mechanics and Mining Sciences 37, 902–922.

Forrester, A.I.J., Keane, A.J., 2009. Recent advances in surrogate-based optimization. Progress in Aerospace Sciences 45, 50–79, http://dx.doi.org/10.1016/j.bbr.2011.03.031.

Frey, H., Patil, S., 2002. Identification and review of sensitivity analysis methods. Risk Analysis 22, 553–578.

Friedman, J.H., 1991. Multivariate adaptive regression splines. The Annals of Statistics 19, 1–67.

Fu, P., Johnson, S.M., Carrigan, C.R., 2012. An explicitly coupled hydro-geomechanical model for simulating hydraulic fracturing in arbitrary discrete fracture networks. International Journal for Numerical and Analytical Methods in Geomechanics , http://dx.doi.org/10.1002/nag.2135.

Goel, T., Haftka, R., Shyy, W., Queipo, N., 2007. Ensemble of surrogates. Structural and Multidisciplinary Optimization 33, 199–216, http://dx.doi.org/10.1007/s00158-006-0051-9.

Hareland G.I., Rampersad, P., Dharaphop, J., Sasnanand S., 1993. Hydraulic fracturing design optimization. In: SPE Eastern Regional Conference and Exhibition, Pittsburgh, PA, USA, SPE Paper 26950, pp. 493–500.

Hsieh, H., 2007. Application of the PSUADE tool for sensitivity analysis of an engineering simulation. Lawrence Livermore National Laboratory, UCRL-TR-237205.

LaPointe, P.R., Hudson, J.A., 1985. Characterization and interpretation of rock mass joint patterns. Geological Society of America Special Paper, 37 pp.

LaPointe, P.R., 1988. A method to characterize fracture density and connectivity through fractal geometry. International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts 25, 421–429.

Long, J.C.S., Witherspoon, P.A., 1985. The relationship of the degree of interconnection to permeability of fracture networks. Journal of Geophysical Research: Solid Earth 90 (B4), 3087–3098.

Mandelbrot, B.B., 1982. The Fractal Geometry of Nature. W. H. Freeman, New York, NY p. 468.

McKay, M., Beckman, R., Conover, W., 1979. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21 (2), 239–245.

Mohaghegh, S., Balanb, B., Platon, V., Ameri, S., 1999. Hydraulic fracture design and optimization of gas storage wells. Journal of Petroleum Science and Engineering 23, 161–171.

Odling, N.E., 1992. Network properties of a two-dimensional fracture pattern. Pure and Applied Geophysics 138, 95–114.

Odling, N.E., 1997. Scaling and connectivity of joint systems in sandstones from western Norway. Journal of Structural Geology 19, 1257–1271.

Picard, R.R., Cook, R.D., 1984. Cross-validation of regression models. Journal of the American Statistical Association 79, 575–583.

Powell, M.J.D., 2009. The BOBYQA algorithm for bound constrained optimization without derivatives. Report DAMTP 2009/NA06, Centre for Mathematical Sciences, University of Cambridge, UK.

Queipo, N.V., Verde, A.J., Canelon, J., Pintos, S., 2002. Efficient global optimization for hydraulic fracturing treatment design. Journal of Petroleum Science and Engineering 35, 151–166.

Queipo, N.V., Haftka, R.T., Shyy, W., Geol, T., Vaidyanathan, R., Tucker, P.K., 2005. Surrogate-based analysis and optimization. Progress in Aerospace Sciences 41, 1–28, http://dx.doi.org/10.1016/j.paerosci.2005.02.001.

Ralph, W., VeatchJr.W., 1986. Economics of fracturing: some methods, examples and case studies. In: SPE 61th Annual Technical Conference and Exhibition, New Orleans, Atlanta, USA, SPE Paper 15509, pp. 1–16.

Renshaw, C.E., 1999. Connectivity of joint networks with power law length distributions. Water Resources Research 35 (9), 2661–2670.

Reeves, D.M., Benson, D.A., Meerschaert, M.M., 2008. Transport of conservative solutes in simulated fracture networks: 1. Synthetic data generation. Water Resources Research 44, http://dx.doi.org/10.1029/2007WR006069.

Rueda, J.I., Rahim, Z., Holditch, S.A., 1994. Using a mixed integer linear programming technique to optimize a fracture treatment design. In: SPE Eastern Regional Meeting, Charleston, South Carolina, USA, SPE Paper 29184, pp. 233–244.

Sanchez, E., Pintos, S., Queipo, N., 2008. Toward and optimal ensemble of kernel-based approximations with engineering applications. Structural and Multidisciplinary Optimization 36, 247–261, http://dx.doi.org/10.1007/s00158-007-0159-6.

Sobol', I, 1993. Sensitivity estimates for non-linear mathematical models. Mathematical Modeling and Computational Experiment 4, 407–414.

Sun, Y., Tong, C., Duan, Q., Buscheck, T.A., Blink, J.A., 2012. Combining simulation and emulation for calibrating sequentially reactive transport systems. Transport in Porous Media 92 (2), 509–526, http://dx.doi.org/10.1007/s11242-011-99170-4.

Tong, C., 2009. PSUADE user's manual (Version 1.2.0). Lawrence Livermore National Laboratory, LLNL-SM-407882.

Walsh, J., Watterson, J.J., 1993. Fractal analysis of fracture pattern using the standard box-counting technique: valid and invalid methodologies. Journal of Structural Geology 15, 1509–1512.

Wang, G.G., Shan, S., 2007. Review of metamodeling techniques in support of engineering design optimization. Journal of Mechanical Design 129 (4), 370–-380, http://dx.doi.org/10.1115/1.2429697.

Wemhoff, A.P., Hsieh, H., 2007. TNT Prout-Tompkins kinetics calibration with PSUADE. Lawrence Livermore National Laboratory, UCRL-TR-230194.

Williams, C.K.I., Rasmussen, C.E., 1996. Gaussian process for regression. In: Touretzky, D.S., Mozer, M.C., Hasselmo, M.E. (Eds.), Advances in Neural Information Processing Systems, 8. MIT Press, Cambridge, pp. 514–520.

PROCEEDINGS, Thirty-Sixth Workshop on Geothermal Reservoir Engineering
Stanford University, Stanford, California, January 31 - February 2, 2011
SGP-TR-191

EFFECT OF UNCERTAINTIES OF GEOMECHANICAL PROPERTIES AND FRACTURE ORIENTATIONS ON FAULT ACTIVATION

Souheil M. Ezzedine[1,2], Joseph P. Morris[3], Lee G. Glascoe[1],
Laura Chiaramonte[4], Tarabay H. Antoun[4], Walter W. McNab[4]

[1]Lawrence Livermore National Laboratory; National Security Engineering Division, L-188
7000 East Avenue, Livermore, CA 94550; e-mail: ezzedine1@llnl.gov
[2]Stanford University, Department of Civil and Environmental Engineering,
396 Via Ortega, Y2E2 Bldg, Stanford, CA 94305
[3]Schlumberger-Doll Research, One Hampshire St, MD-B408, Cambridge, MA 02139, USA
[4]Lawrence Livermore National Laboratory; National Security Engineering Division, L-286
7000 East Avenue, Livermore, CA 94550

ABSTRACT

Predicting the ultimate fate of the injected of water or supercritical carbon dioxide in the subsurface for storage or heat extraction involves understanding the interrelationship between multiple processes, such as the hydrological, mechanical and chemical transport of the injected working geofluid. The majority of these processes take place within the fracture and faults which may lead to compromising the integrity of the reservoir. Results, obtained using LDEC, which analyze the coupling of fluid flow and stresses within extensive combined fracture-fault networks are presented. Moreover, subsurface is inheritably heterogeneous and hydromechanical properties can vary spatially from one location to another. Thus, a second analysis has been conducted assuming that the geomechanical properties are randomly generated between both the weak and strong cases. Uncertainty with the fracture network orientation could also impact the large scale response and activation of the faults and thus the integrity of the reservoir, the containment of the working geofluid, the surface deformation and ultimately the fluid-induced seismicity. A third analysis was conducted based on alteration of the orientation of the fracture network from the base case scenario. The resulting effect on the friction and activation conditions along the main faults is assessed.

INTRODUCTION & PROBLEM STATEMENT

Global temperatures are increasing. Atmospheric CO_2 concentration is increasing. New carbon management legislation and initiatives are based on the assumption that increased CO_2 levels are a major cause of global warming. Several ways of reducing CO_2 emissions and CO_2 levels in the atmosphere include: improved efficiency in power generation by upgrading existing plants, higher efficiency in all new plants, relying more on renewable energy such as enhanced geothermal system (EGS, Figure 1a) using water or supercritical CO2 as a geofluid, and finally, carbon capture and storage (CCS, Figure 1b). EGS and CCS share similar technological fundamentals and the consequences of their deployment on the environment as well as on the public perception such as induced seismicity.

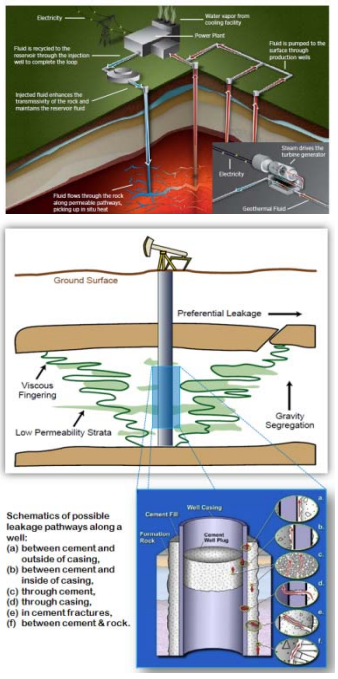


Figure 1: a) [Top] Basics of an Enhanced Geothermal (Adapted from eere.energy.gov); b) [Bottom] Basics of Geological Carbon Sequestration and Well Integrity (Adapted from Carbon Sequestration Research and Development, 1999, Chapter 5and Gasda et al., 2004)

Undoubtedly, predicting the ultimate fate of the injected water or supercritical carbon dioxide in the subsurface for storage or heat extraction involves understanding the interrelationship between multiple

processes, such as the hydrological, mechanical, thermal and chemical transport of the injected working geofluid (Figure 2, Ezzedine, 2008). The majority of these processes take place within the fracture and faults.

Figure 2: Potential Hydrological, Thermal, Chemical and Mechanical coupling affecting the design of a geothermal or a geological carbon sequestration system.

For instance, reactions induced by the presence of geofluid and changes in stress state due to large volume of injected geofluid will result in significant changes in the permeability of the fractures through hydro-thermo-mechanical and chemical precipitation and dissolution, healing/precipitation, which alter the permeability tensor within the reservoir. Changes may be relatively small, but their impact could be large. Furthermore, the large rate and volume of injection will induce pressure and stress gradients within the formation that may activate existing fractures and faults, or drive new fractures through the caprock, creating new fractures, and thus compromising the integrity of the reservoir. We will report results obtained using LDEC to analyze the hydromechanical coupling of fluid flow and stress within extensive combined fracture-fault networks. We will also discuss the implications these results have for the transport and ultimate fate of the working geofluid. Moreover, subsurface is inheritably heterogeneous and hydromechanical properties can vary spatially from one location to another. Therefore, a second analysis has been conducted assuming that the geomechanical properties, for instance, are randomly generated between both the weak and strong cases. Uncertainty with the fracture network orientation could also impact the large scale response and activation of the faults and thus the integrity of the reservoir, the containment of the working geofluid, the surface deformation (uplifting) and ultimately the fluid-induced seismicity. A third analysis was conducted based on alteration of the orientation of the fracture network from the base case scenario. The resulting effect on the friction and activation conditions within the main faults is assessed.

THE SIMULATION TOOL: LDEC

The massively parallel Livermore Distinct Element Code (LDEC) is deemed to be the appropriate tool for conducting the proposed analyses (Morris et al., 2002). LDEC is an implementation of the distinct element method (DEM) (Cundall, 1988). Using this approach one can directly approximate the block structure of the jointed rock using arbitrary polyhedra. Preexisting joints are, therefore, readily incorporated into the DEM model. Furthermore, the distinct element method can readily handle large deformation on the joints. In addition, the method detects all new contacts between blocks resulting from relative block motion. The Lagrangian nature of the DEM simplifies tracking of material properties as blocks of material move. Furthermore, it is also possible to guarantee exact conservation of linear and angular momentum. By using an explicit integration scheme, the joint models can be very flexible. The joint constitutive models in LDEC include effects such as, cohesion, joint dilation, and fiction angle to name a few. LDEC implements two approximations to block response: rigid and deformable (Morris et al., 2002).

SIMULATED SCENARIOS

First, we have conducted strong and weak hydromechanical scenarios and compare them to the base case scenario reported in Morris (2009) (see Figure 3 for the distribution of fractures and faults). Three cases where identified based on the Young modulus E: 1) weak case: E=5GPa, 2) mild (base) case: E=10GPa; and 3) strong case: E=20GPa. The Poisson's ratio is set to 20% for all cases. For the current analyses, only Young modulus has been changed, Poisson's ratio remained constant throughout all subsequent analyses.

These three analyses enable us assessing the activation of the faults under different ultimate conditions and not relying solely on a base case scenario. Subsurface reservoirs are inheritably heterogeneous; therefore, the hydromechanical properties can vary spatially from one location to another. Thus, a second analysis has been conducted assuming that the geomechanical properties, for instance, are randomly generated between both the weak and strong cases. Uncertainty with the fracture network orientation could also impact the large scale response and activation of the faults and thus the integrity of the reservoir, the containment of the injected water or CO2, the surface deformation

(uplifting) and ultimately the fluid-induced seismicity. A third analysis was conducted based on shifting the fracture network by 25 degrees from the base case scenario. The resulting effect on the friction conditions within the main faults is assessed.

RESULTS

Simulations involve an extensive fracture networks (about 400 thousands of fractures), including detailed intersections with faults (four faults). We consider only fractures and faults within the reservoir itself. These faults have not been observed to persist into the overburden. Consequently, this initial study is concerned only with assessing impact of different geomechanical conditions on the evolution of the reservoir and not the caprock response.

Base Case Scenario

Figure 3 shows the fault-fracture network for the Krechba reservoir built using LDEC using the same assumptions previously reported in Morris (2009). Figure 4 depicts the response of the combined fracture and fault network to pressure increase overlaid by the slip conditions on the faults.

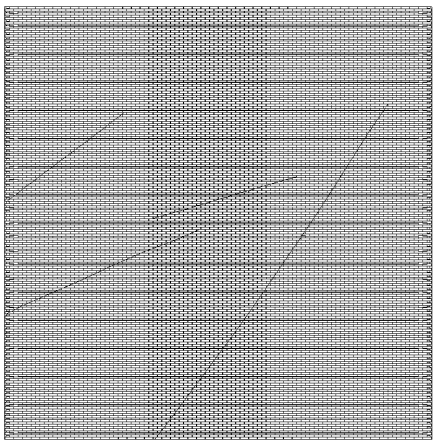


Figure 3: Fracture-fault network model for the Krechba reservoir built within LDEC, highlighting the details of the individual fractures within the network. The model includes 400,000 individual fracture elements.

Pressure field is chosen to reflect conditions observed on site. Throughout the entire document all figures are plotted for same fluid pressure conditions of CO2 in the reservoir in order to facilitate the comparison between the different cases. This calculation considers the poroelastic response of the fractured rockmass and includes the redistribution of stresses through the combined fracture-fault network.

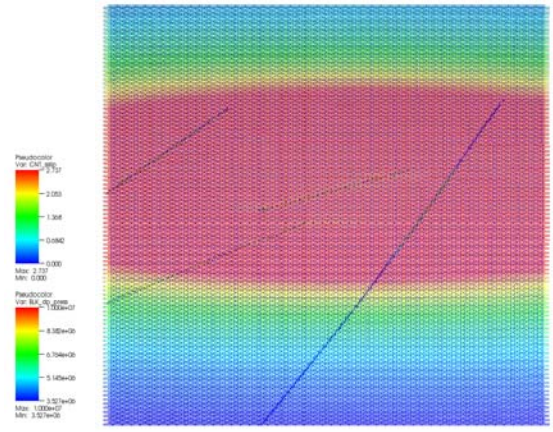


Figure 4: Overlay of the changes in the pressure field and the slip conditions on the fault.

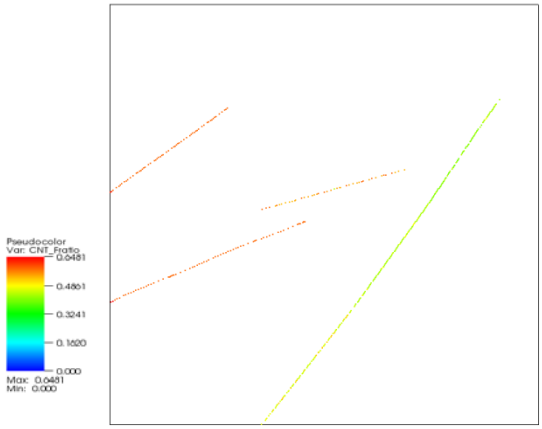


Figure 5: propensity for shear on the faults. The color scale reflects the coefficient of friction required to maintain stability of the segments of the faults.

Figure 5 shows our LDEC prediction of propensity for shear on the faults. The color scale reflects the coefficient of friction required to maintain stability of the segments of the faults. As faults approach a ratio of 0.6, they are presumed to become permeable. Consequently, it can be seen that several faults are critically stressed and likely to provide fast flow paths.

Weak Case

Analyses conducted in previous section have been re-analyzed for a weak case scenario where the bulk modulus is set to 5GPa and Poisson's ratio fixed at 20%. Figure 6 depicts, for the same poroelastic conditions as in the base case scenario, propensity for shear on the faults. Not surprisingly, weaker conditions promote less stability and more slips on the fault. One can

also draw the conclusion that the activated fault surfaces is by large greater than for the base case and confirmed on Figure 7 which depicts the slip conditions on faults.

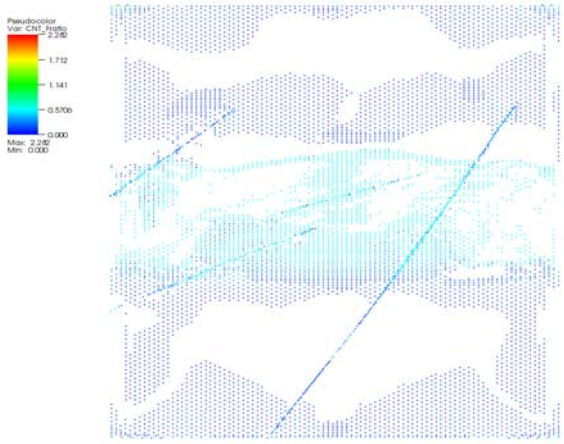


Figure 6: Weak case scenario: propensity for shear on the faults. The color scale reflects the coefficient of friction required to maintain stability of the segments of the faults.

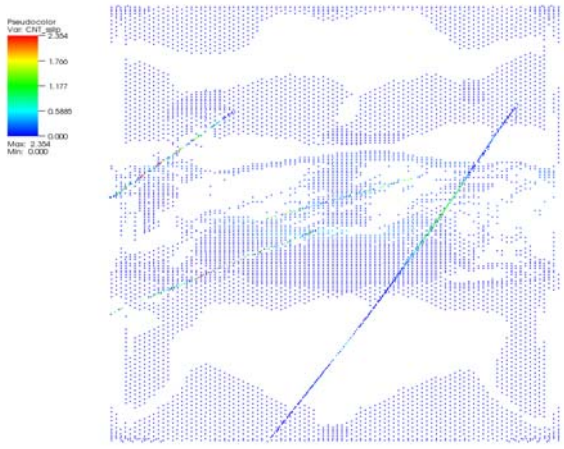


Figure 7: Weak case scenario: slip conditions on the fault.

Strong Case

Similarly to the weak case, same re-analysis has been conducted for a stronger case scenario where the bulk modulus is set to 20GPa and Poisson's ratio fixed at 20%. Figure 8 depicts, for the same poroelastic conditions as in the base case scenario, propensity for shear on the faults. Stronger conditions promote more stability and less slips on the fault. Figure 9 depicts the slip conditions on faults. It is worth noting that the activated fault surfaces are less than those calculated in the base case scenario.

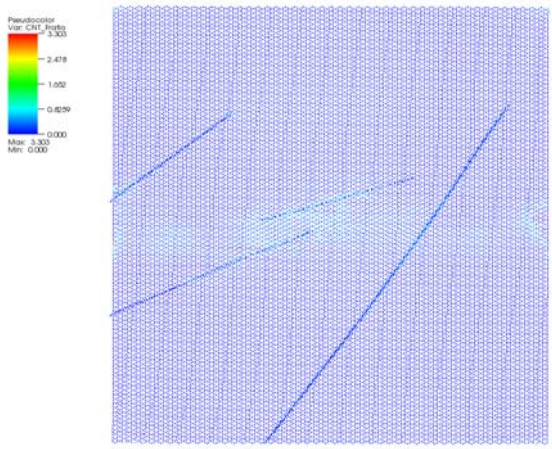


Figure 8: Strong case scenario: propensity for shear on the faults.

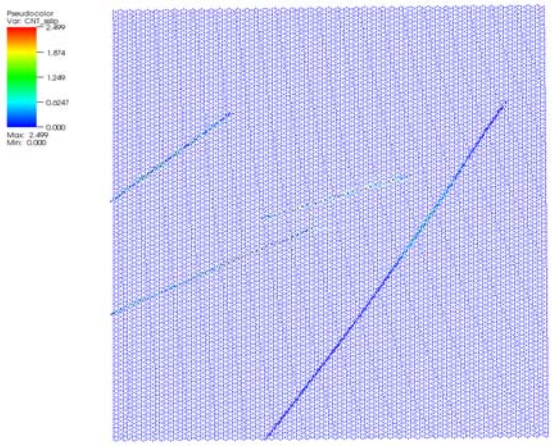


Figure 9: Weak case scenario: slip conditions on the fault.

Random Geomechanical Properties

Subsurface conditions are by nature heterogeneous; therefore, one can expect that the subsurface poroelastic properties vary in space from one location to another. Moreover, analyses conducted in the previous subsections assume that the properties are not only strongly homogeneous but also take only extreme values (e.g. weak and strong). The coexistence of weak, average and strong poroelastic properties is the rule and not the exception; therefore their simultaneous signatures in the subsurface need to be assessed through probabilistic framework to predict, for example, the impact of their spatial variabilities and uncertainties on the activation of the faults and thus the integrity of the reservoir. As a first step toward this goal, an uncorrelated random fields of the density as well the Young's modulus of the rock in the reservoir have been generated and unconditionally

assigned to the rock mass. Figures 10 and 11 depict the density and Young's modulus random field, respectively. In average, both fields reflect the base case scenario. It is worth noting however, that in reality, damaged zone (area) are generally localized around the faults themselves, or for instance, the zone between the two central faults (see Figure 3) may have more damaged rock mass and thus weaker properties localized between them. Moreover, the correlation between the different subsurface properties calls for conditional simulations as the probabilistic tool. For illustration purposes, we limit ourselves to single unconditional case as a first step toward building a stochastic framework within LDEC.

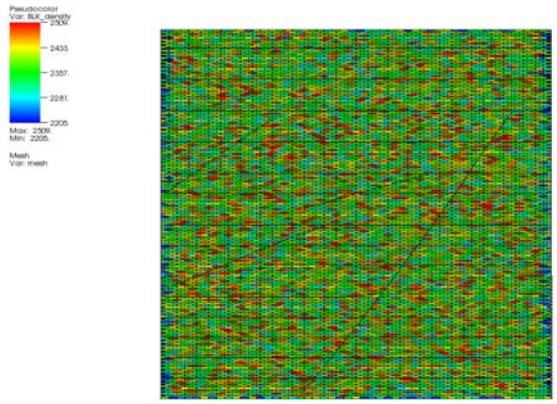


Figure 10: Example of uncorrelated random of the density of the rock of the reservoir.

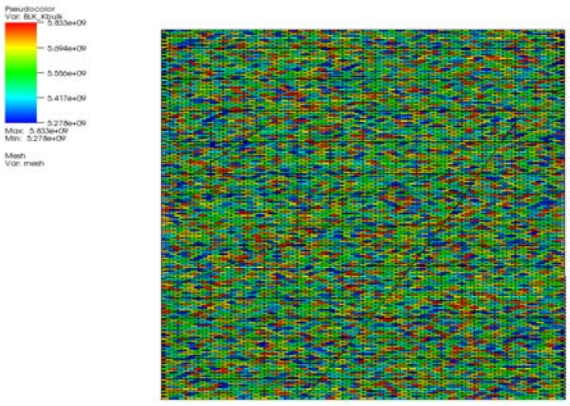


Figure 11: Example of uncorrelated random of the bulk modulus of the rock of the reservoir.

Similarly to the base case, all analyses have been re-conducted for this single random case. Figures 12 and 13 depict the shear slip conditions and propensity for shear, respectively. Not surprisingly, the results are somewhat a mixture of the weak, base and strong cases conducted in the previous subsections. It is however important to mention that this is the outcome of a single realization and an average though a series of realization is needed.

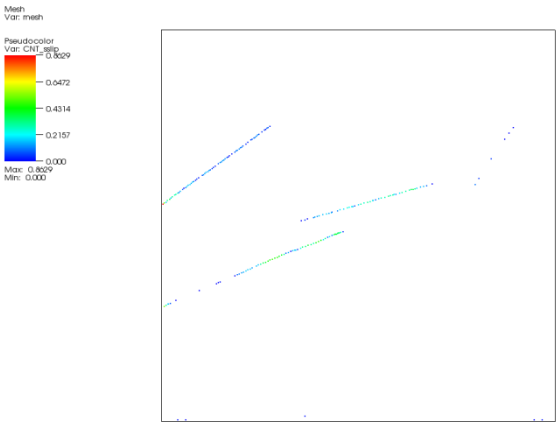


Figure 12: Slip conditions for the random properties case

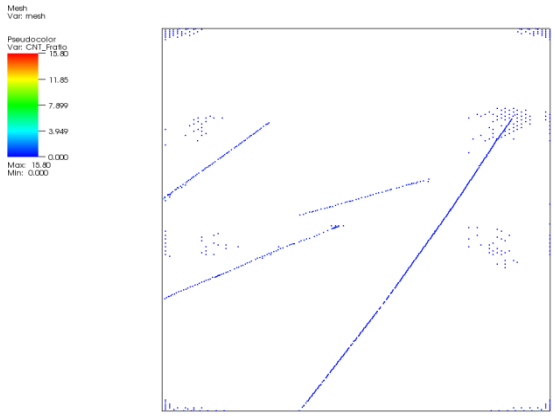


Figure 13: Propensity for shear on the faults for the random property case

Impact of Fracture Orientation

Rock mass properties are not only the only uncertain, spatially variable parameters in the subsurface, fracture (joint) orientation is another one (Ezzedine, 2010). Without further ado, and for illustration purpose, we have re-conducted the analyses on a fracture network geometry that is 25 degree of the base case scenario. Figure 14 depicts the fracture-fault network build for the current analysis. Only the fractures are re-oriented to 25 degree off the base case scenario, major fault orientation remained the same. The computational overburden remains almost the same compared to the base case analysis. This geological setting of the reservoir has been simulated to assess the effect of the fracture

orientation on the activation of the faults. It is worth noting that the in-situ stresses and orientations remain unchanged with respect to the previous analyses. Figure 15 shows propensity for shear on the faults. Because the fracture network is at 25 degrees the larger fault is more potent to shear practically along the entire fault, which is not observed throughout the previous analyses. Furthermore, the second major fault is well aligned with the new orientation of the fracture network and thus more shear slip is experienced (larger values, see legend of Figure 15). In addition, the motion on the faults leads to stress perturbations that induce shear loading on adjacent fractures. For this specific scenario, the fractures have not failed at this level of pressure perturbation. However, this coupled interaction between the faults and fracture network must be considered when predicting the permeability evolution within the reservoir (Ezzedine, 2005).

Again this analysis is limited and a more thorough investigation needs to be conducted assuming more realistic random network of fractures and uncertainty around the orientation of the main family of fractures.

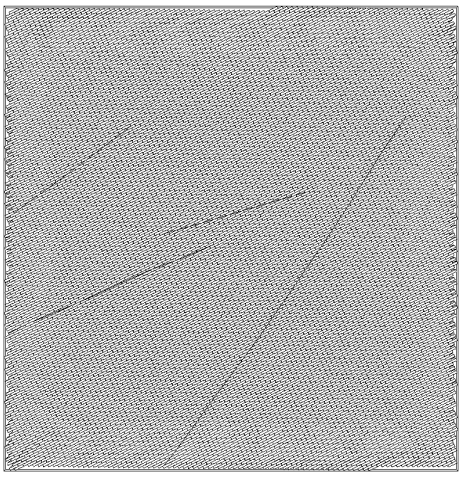


Figure 14: Fracture-fault network model for the Krechba reservoir built within LDEC, highlighting the details of the individual fractures within the network. The model includes 400,000 individual fracture elements. Notice that the fractures are 25 degree off from those presented on Figure 3. Fault orientation remains unchanged.

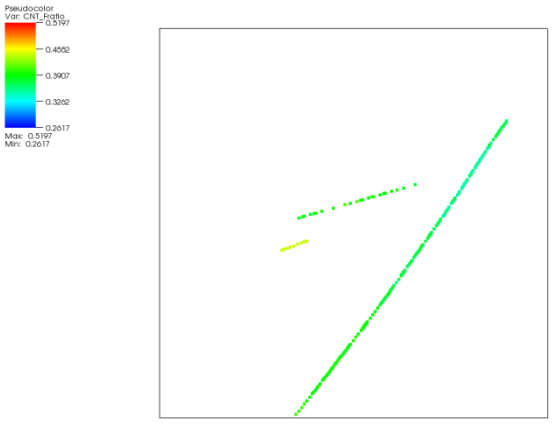


Figure 15: Propensity for shear on the faults for the re-oriented fracture network case

CONCLUSIONS AND PATH FORWARD

We have attempted, through a series of numerical simulations using LDEC, to predict the behavior of the faulted system and the response of the overall reservoir under poroelastic conditions similar to In-Salah's. We have also built a stochastic framework with LDEC to assess the impact of uncertainty not only in the physical (geomechanical) properties but also geological uncertainty due to the characterization of the orientation the fracture network. Results presented here are preliminary and more numerical simulations need to be conducted based on correlated and conditional realization of the subsurface at In-Salah.

Future work will include revising this model to include more fractures and varied fracture sets. The model will be iteratively coupled with reservoir scale reactive transport modeling to investigate the interaction between mechanical and geochemical processes. In-situ stresses are by far the most uncertain in a geomechanical uncertainty assessment. Undoubtedly, the current stochastic framework can readily be extended to include such uncertainty. The outcome of this framework will serve as an input to an integrated risk assessment and management of the reservoir to minimize its integrity breach, minimize surface disturbances and mitigate any induced seismicity that may occur due to the injection of geofluids.

REFERENCES

Cundall, P. A. Formulation of a Three-Dimensional Distinct Element Model — Part I: A Scheme to Detect and Represent Contacts in a System Composed of Many Polyhedral

Blocks," Int. J. Rock Mech., Min. Sci. & Geomech. Abstr., 25, 107-116 (1988).

DOE, How an Enhanced Geothermal System Works, eere.energy.gov/geothermal/pdfs/egs_basics.pdf, 2011.

DOE, Carbon Sequestration Research and Development, 1999, www.fossil.energy.gov/programs/sequestration/publications/1999_rdreport/. Chapter 5. 1999.

Ezzedine S., Stochastic Modeling of Flow and Transport in Porous and Fractured Media. *Chapter 154 in Encyclopedia of Hydrological Sciences*, 34pages, Willey 2005

Ezzedine, S., Coupled THMC processes in Geological Media using Stochastic Discrete Fractured Network. Application to HDR Geothermal Reservoirs. *Eos Transactions AGU*, 89(53), Fall Meeting Supplement, Abstract H41A-084, 2008

Ezzedine, S., A comprehensive Uncertainty Quantification in Enhanced Geothermal Systems using Stochastic Discrete Fractured Network. A Computational Approach. Submitted to *Computational Geosciences*, 2010.

Gasda, S.E., S. Bachu, and M.A. Celia, "The Potential for CO2 Leakage from Storage Sites in Geological Media: Analysis of Well Distribution in Mature Sedimentary Basins", Environmental Geology, 46 (6-7), 707-720, 2004.

Morison, J.P., Simulations of Injection-Induced Mechanical Deformation: A Study of the In Salah CO2 Storage Project. SEG 2009 Summer Research Workshop on CO2 Sequestration Geophysics, 23-27 August 2009, Banff Canada.

Acknowledgments:

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-PROC-422871-DRAFT.

PROCEEDINGS, Thirty-Eighth Workshop on Geothermal Reservoir Engineering
Stanford University, Stanford, California, February 11-13, 2013
SGP-TR-198

UNCERTAINTY QUANTIFICATION OF REACTIVE TRANSPORT IN FRACTURED MEDIA

Souheil Ezzedine and Fredrick Ryerson

Lawrence Livermore National Laboratory
7000 East Avenue, Mail Stop L-188
Livermore, CA, 94550, USA
e-mail: ezzedine1@llnl.gov

ABSTRACT

Uncertainties in elementary reaction rates of chemical mechanisms have traditionally been reported by experimentalists. The development of methodologies to represent the chemical reaction-rate measurement uncertainties is a necessary step towards quantitative predictions of complex reactive transport in fractured media. The method presented in this study provides a useful framework for quantifying uncertainties in both mineral surface area and activation energies. Brute force Monte-Carlo simulations of flow, heat and mass transport in a single fracture illustrating the propagation of the chemical and the physical characterization uncertainties and their impacts on the aperture changes of the fracture under different boundary conditions.

INTRODUCTION

The chemical processes of solute transport are governed by the principles of conservation of mass, energy, and charge. All chemical processes in closed systems under constant conditions tend toward equilibrium by the second law of thermodynamic, but the speed of approach to equilibrium varies widely. Therefore, almost all current modeling efforts distinguish between two broad classes of chemical processes: 1) those which are sufficiently fast and reversible, so that local chemical equilibrium may be assumed to exist, and 2) those which are insufficiently fast and/or irreversible reactions, where the local equilibrium formulation is inappropriate.

Rubin [1983] has two further levels of classification of the chemical reactions within each of two above classes (other classification schemes are also possible). The first distinguishes reactions by the number of phases involved, either homogeneous (with a single phase) or heterogeneous (more than one phase). Second, within the heterogeneous class of reactions are the surface reactions (adsorption and ion exchange reactions), and the classical chemical reactions (precipitation/dissolution, oxidation or redox reaction, and complex formation, although redox and complex formation are not always heterogeneous). There are three types of chemical reactions that are considered significant for chemical transport: ion complexation in the aqueous solution (fluid phase), sorption on solid surfaces (interphase boundaries), and precipitation/dissolution of solids (solid phase).

For the propose of deriving the chemical transport equations, we assume that the aqueous component species and complex species are subject to hydrological transport, whereas the precipitated species absorbent component species, adsorbed species, and ion-exchange species are not subject to hydrological transport. The general transport equation governing the temporal-spatial distribution of any chemical species in multicomponent system can be derived based on the principle of conservation of mass.

Non-Equilibrium Transport Formulation

In line with Miller et al. (1983) and Noorishad et al. (1987), the partial differential equation of dispersive-advective mass transport does not change fundamentally for consideration of non-equilibrium chemical reactions. If basis species are produced or consumed irreversibly at finite rate a source term must be added to the equilibrium transport equation for each basis species involved in the irreversible reaction. The mass balance equation for chemical transport for any species ‘j’ can be written as:

$$\mathcal{L}\left[c_{bj}+\sum_{i=1}^{Nc}a_{ij}c_{ci}\right]+\frac{\partial}{\partial t}\left[c_{bj}+\sum_{i=1}^{Nc}a_{ij}c_{ci}\right]+$$
$$\frac{\partial}{\partial t}\left[\bar{c}_{bj}+\sum_{i=1}^{\bar{N}c}a_{ij}\bar{c}_{ci}\right]+\sum_{i=1}^{Np}s_{ij}\frac{\partial c_{pi}}{\partial t}+\delta_{kj}\nu_{kb}R_b=0,$$
$$\forall j=1,\cdots,Nb$$

Where $\mathcal{L}$ denotes the advection-dispersion spatial operator (div(D gard[] - V[]), c_{bj} is the concentration of basis ion "j" [mol/l], \bar c_{bj} is the concentration of sorbed basis ion "j" [mol/l] of fluid phase, c_{ci} is the concentration of complex ion "i" [mol/l], $\underline{c}_{ci}$ is the concentration of sorbed complex ion "i" [mol/l] of fluid phase, c_{pi} is the concentration of solid "i"[mol/l] of fluid phase, a_{ij} are the stoichiometric coefficients for formation of complex "i", s_{ij} are the stoichiometric coefficients for formation of solid "i", V is the fluid velocity vector [m/s], D is the dispersion coefficient [m^2/s], N_b is the number of basis species, $\underline{N}_b$ is the number of sorbed basis species, N_c is the number of aqueous complexes, $\underline{N}_c$ is the number of sorbed complexes, N_p is the number of solids. The second term of the left hand is the rate of accumulation of the "j"th substance due to aqueous phase reactions (mainly ion complexation), the third is similarly a term for sorption, and the forth for precipitation and dissolution. The fifth term is the reaction term where k refers to basis species involved in the reaction and has one or more discrete values in the range 1,…,N_b. δ_{kj} is the Kronecker delta function, ν_{kb} is the stoichiometric coefficient of basis species "k" in the reaction, and R_b is the rate of reaction. On the other hand, if complexes or solids participate in non-equilibrium reactions and if no mass enters or leaves the system irreversibly, then the equilibrium equation is used without modification. However, the mass action equations for each participating complex or solid must be replaced by rate equations of the type: $\partial X_i/\partial t = - R_{Xi}$ where X_i represents the concentration of a complex or a solid and R_{Xi} is the correspondent rate. The mass action for non-participating complexes and solids remain unchanged.

Chemical Equilibrium Reactions

Species distribution can be formulated in two distinct but thermodynamically related ways: the equilibrium constant approach and the Gibb's free-energy approach. In the Gibb's free-energy approach, the species distributions are obtained by minimizing the total Gibb's free-energy function of a given set of species subject to the constraints of mass balance equations. In the equilibrium constant approach, the set of nonlinear algebraic equations (AEs) is obtained based on the law of mass action and the principle of the mole balance (Morel, 1983). This set of nonlinear AEs is then solved to yield the species distributions. In the latter approach, equilibrium constants are needed, whereas in the former approach, free-energy values are needed. In the equilibrium constant approach, the formation of complex species, adsorbed species, ion-exchanged species, or precipitated (solid) species is described at equilibrium by the mass action laws. These mass action laws are not given here because they are classic and available in textbooks (Morel, 1983).

NUMERICAL SIMULATION FRAMEWORK

The proposed model of reactive transport of species has been implemented in the SDFN-THMC framework (Ezzedine, 2005) and has been extensively used in recent years to address the impact of uncertainties in the geological characterization of fracture on the thermal response of an EGS (see Ezzedine, 2009-2011). It was originally coded to simulate the system at Soultz-sous-Forêts. Fractures are either deterministic or stochastic. Fractures are characterized by their density, orientation, size and aperture. The model allows for multiple sets of fractures, each having their own probability distribution (density) functions. The model is equipped with several numerical schemes for solving the different previously mentioned processes and different protocol for the numerical coupling of those processes (see Figure 1) Moreover, it offers different geological conceptualization of the fractures and how the physical processes are solved within each fracture of the fracture network.

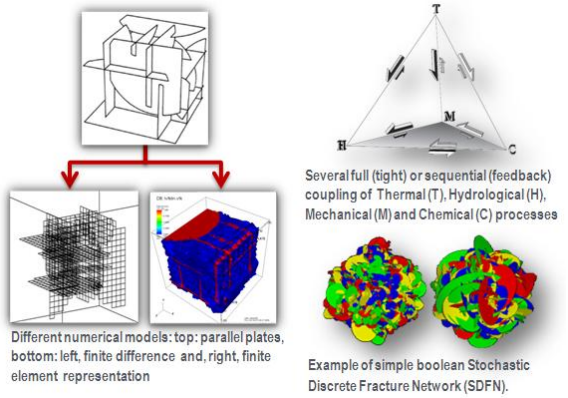


Figure 1: SDFN-THMC (Ezzedine, 2005) offers an ideal numerical framework to address the problem of deploying silica gel in EGS. The model allows for flow, thermal, and mechanical & chemical simulation in a 3d stochastic discrete fracture network.

Numerical Algorithms

Yeh and Tripathi (1989) describe three basic methods for solving the solute transport and chemical equations. The first approach is to solve the transport and chemical equilibrium equations simultaneously without substitution (NUFT, Nitao 1998, Miller and Benson. 1983, and Noorishad et al. 1987). The second is the direct substitution approach (DSA), which involves substituting the chemical equilibrium equations into the transport equations. This approach was used by Rubin and James 1973, Valocchi et al. 1981, Jennings et al. 1982, Rubin 1983, and Lewis et

al. 1987. The third approach involves splitting the calculation into transport equation solution step and chemical equilibrium calculation step and performing the two steps sequentially (ToughReact, Xu and Pruess, 1998). The transport equations are solved individually for the total aqueous concentration of every component. This approach was used by Grove and Wood 1979, Walsh et al 1984, Kirkner et al. 1984, 1985, Cederberg et al. 1985, Yeh and Tripathi 1989, 1991, and Kinzelbach and Schafer 1989. Some of these models iterate between the transport and chemical equilibration calculations at each time step. The third approach with iteration is called the sequential iteration approach (SIA). In SDFN-THMC we opted with one-step DAE solution which is numerically robust at the expenses of large computer memory requirements.

Effect of heat transport on chemical reactions

The formulations presented so far are generally based on the assumption of isothermal conditions. However, considering the role of variation of temperature in any geochemical processes, it is necessary to include thermal effects in chemical reactions and to formulate heat transport as a companion to mass transport. Removing the restriction of constancy of temperature does not alter the mass transport formulation in any way. The temperature change effect is accommodated by changing the equilibrium constants. These are computed at any desired temperature (within the range of data validity) from a power function such as [Reed,1982]:

$$\log_{10}[K] = A + B\,T + C\,T^2 + D\,T^3 + E\,T^4$$

where K is the thermodynamic equilibrium constant used in any mass action relation. A, B, C, D and E are parameters that vary for different chemical reactions, and T is the absolute temperature.

Solution Methods

As discussed earlier, time and space discretization of the governing equations for both reactive chemical transport and heat transport lead to matrix equations. For heat transport we have used the Crank-Nicholson approximation rather than fully implicit methods; however, the final form is the same. In case of reactive chemical transport equations, the strong nonlinearity of the matrix equations mandates the use of an iterative scheme. We have used a generalized Newton-Raphson method.

APPLICATION TO CALCITE DISSOLUTION

Plummer et al. (1978) has conducted a seminal study of the dissolution of calcite in CO_2-H_2O solutions over pH ranges from about 2-7, PCO_2 ranges from 3E-4 to 0.97 atm and temperature from 5 to 60^{o}C. They proposed the simultaneous occurrence of the following three reactions on the calcite surface:

$$Ca\,CO_3 + H+ = Ca^{2+} + HCO_3^- \quad (R1)$$
$$Ca\,CO_3 + H_2CO_3^o = Ca^{2+} + 2HCO_3^- \quad (R2)$$
$$Ca\,CO_3 + H_2O = Ca^{2+} + OH^- \quad (R3)$$

The net rate of dissolution, R, is given by:

$$R=d[Ca^{2+}]/dt=k_1a_{H+}+k_2a_{H2CO3}+k_3a_{H2O}-k_4\,a_{ca2+}\,a_{HCO3-}$$

Where (a_i)s are the thermodynamic activities of the i^{th} reaction and k_1, k_2 and k_3 are the rate constants for reaction R1, R2, and R3, respectively. The backward reaction constant k4 depends on temperature and PCO2 and is given by:

$$k_4 = K_2/K_C\,[k_1' + 1/a_{H+(s)}\,(k_2\,a_{H2CO3} + k_3\,a_{H2O})]$$

where the subscript (s) denotes values on the surface of the calcite, K_2 is the second dissociation constant of carbonic acid and K_C is the calcite equilibrium constant given in Table 1. The first other mechanistic rate constant of H^+, k_1', is 10 to 20 times larger than measured k_1 which is interpreted as the H^+ transport rate constant. The effect of uncertainty of k_1' is addressed in the uncertainty quantification subsection.

Table 1: Thermodynamic data at 25^{o}C for the system $CaCO_3$-CO_2-H_2O

Reaction	Log K	Source
$CaCO_3$ (Calcite) = Ca^{2+}+ CO_3^{2-}	-8.475	JL (1974)[a]
CO_2+H_2O = $H_2CO_3^*$	-1.466	HD (1943)[b]
$H_2CO_3^*$ = H^++HCO_3^-	-6.351	HD (1943)[b]
HCO^{3-} = H^++CO_3^{2-}	-10.330	HS (1941)[c]
$CaHCO_3^+$ = Ca^{2+}+HCO_3^-	-1.015	JL (1974)[d]
$CaCO_3^o$ = Ca^{2+}+CO_3^{2-}	-3.153	RE (1974)[e]
H_2O = H^++OH^-	-14.000	H (1969)[f]
a-Jacobson & Langmuir ('74), b- Harned & Davis ('43), c-Harned & Scholes ('41), d- Jacobson & Langmuir ('74), e- Reardon & Langmuir ('74), f- Helgeson ('69)		

Temperature dependency of reaction constants

The variations of the reaction constant as function of temperatures are given by:

$$\log_{10}[k_1] = 0.198 - 444/T$$
$$\log_{10}[k_2] = 2.84 - 2177/T$$
$$\log_{10}[k_3] = -5.86 - 317/T \quad \text{if } 278^oK \le T < 298^oK$$
$$\log_{10}[k_3] = -1.10 - 1737/T \quad \text{if } 298^oK < T < 321^oK$$
$$\log_{10}[K_C] = -8.446$$
$$\log_{10}[K_1] = -40.366 + 5576.57/T + \ldots + 11.331 \log10\ T + 0.034655\ T$$
$$\log_{10}[K_2] = -77.3723 + 4593.31/T + \ldots + 29.042 \log10\ T + 5.944E\text{-}4\ T$$
$$\log_{10}[K_W] = -80.3723 - 1395.46/T + \ldots$$

…+32.5514 log10 T - 0.0318065 T

For verification and validation purposes we treat here only the one dimensional case. Two and three dimensional cases will be presented in subsequent work. The current exercise was taken from Miller and Benson (1983) for comparison with LBNL 1-step code CHMTRANS. The initial conditions are: P_{CO2} = 3.1623E-2atm, pH=5, C_{H2CO3}=1.473E-3 mol, C_{HCO3-}= 5.637E-5 mol, C_{CO32-}=1.865E-10 mol. The length of the fracture (tube) is set to 10m which was discretized into 20 elements. The pressure boundary conditions were set to yield a Darcy velocity of 1E-3 m/year. Even though this velocity is well below velocities experienced in fractures for example, it serves two purposes: 1) direct comparison with Miller and Benson (1983) and 2) allow enough residence time for dissolution and considerable changes in aperture of the fracture.

The distribution of chemical species at the end of the outlet of the fracture is given on Figures 2-4. Concentrations are given in mol/l and time in years. To further illustrate the impact of the physic-chemical conditions, results were also depicted for transport with equilibrium, transport with kinetics and batch kinetics only (no transport) on the same figures. It is worth noting the irregularity of the curves, it is because we are not interested by high precision but by the accurate representation of the physics and the chemistry that take place in the fracture. We are building a framework to study the uncertainty propagation through a model and its impact and value rather than "a model with the best number". We are targeting consistency and accuracy rather than precision. The results were obtained using k_1'=15. As one can see there are significant differences between the different combinations of processes at hands. The impact of the different transport processes for different species are depicted on Figures 5-7 for the time-evolution of $(CO_3)^{2-}$, Ca^{2+}, and HCO_3^- & H_2CO_3, respectively.

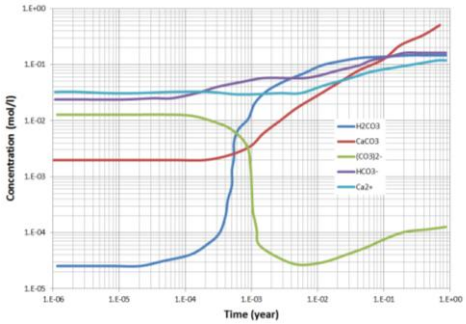


Figure 2: Transport with kinetics. Concentration of species at the end of the fracture.

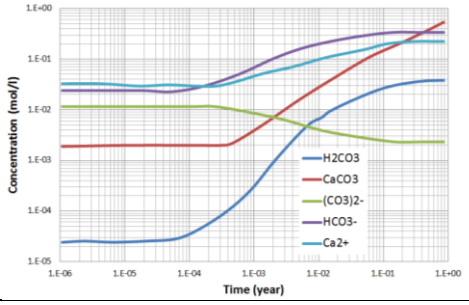


Figure 3: Transport with equilibrium. Concentration of species at the end of the fracture

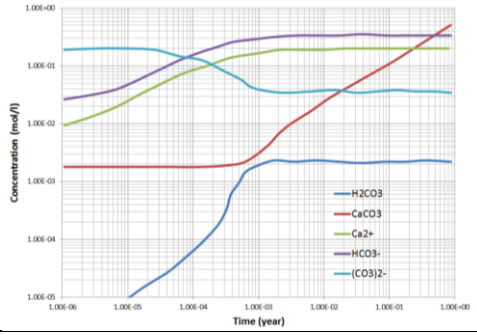


Figure 4: Kinetics without transport. Concentration of species at the "end" of the fracture

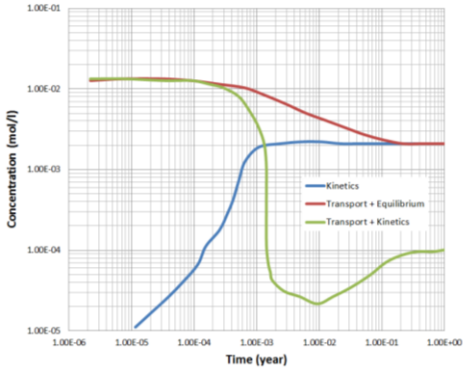


Figure 5: Evolution of $(CO_3)^{2-}$ for different physic-chemical processes

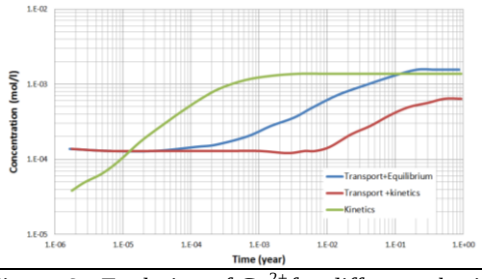


Figure 6: Evolution of Ca^{2+} for different physic-chemical processes

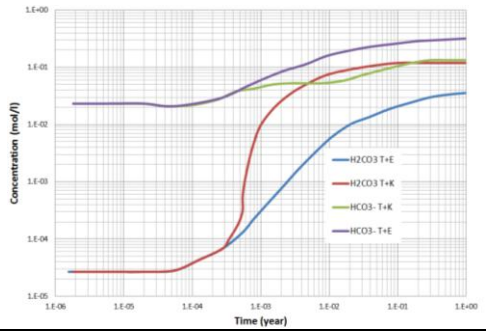


Figure 7: Impact of transport and equilibrium and transport and kinetics for HCO_3^- and H_2CO_3.

DISSOLUTION RATE UNCERTAINTY

Uncertainty within the experimental data sets

The dissolution of carbonate minerals, calcite in particular, has received considerable study in the geochemical kinetics literature (Arvidson et al. 2003, and references therein). Despite the accumulation of a large dataset over several decades, there is significant uncertainty in the value of the dissolution rate under given conditions. This dataset is summarized in Figure 8 (Arvidson et al. 2003).

Calcite dissolution is controlled by surface-reaction (as opposed to transport) kinetics with increasing pH under alkaline conditions far from equilibrium (high under-saturation). Figure 8 shows that although the data exhibit internal consistency within results from a given laboratory, absolute rates at high pH vary by well over an order of magnitude which is surprising given the large quantity of experimental data available. Some of the differences in rate reflect differences in experimental conditions (e.g., ionic strength, P_{CO2}, alkalinity). However, it is difficult to evaluate what additional sensitivity may be present as a function of experimental and analytical methodology, starting materials, or other factors. We are not concerned with the true value but given an ensemble of possible rate law from Figure 8 one can determine the impact of those models on the ensemble average of distribution of the species. If we cannot understand the origin of differences in rates derived from changes in solution chemistry, then neither can we fully understand the relationship of those rates. We thus opt for probabilistic uncertainty quantification through direct Monte Carlo simulations. The rate of dissolutions is on Figure 8 will serve as a driver for the conditional probability of rates given a pH. Here the conditional probability is uniform between the min and max of the possible rates. Furthermore, k_1' is considered as a uniform between [10,20] as stated in previous section. There are several other uncertain factors which could be geological or physical which increase the dimensionality of the problem. We focus here, without loss of generality of scheme, on the uncertainty within the rate of dissolution. It is our ultimate aim to understand how this uncertainty, regardless its origin, can propagate through a THC process.

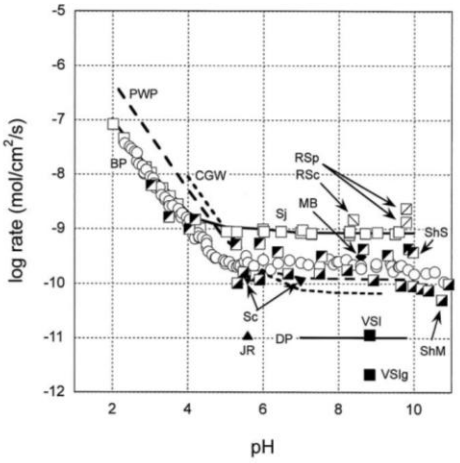


Figure 8: Published rates of calcite dissolution: Solid line with open squares ('Sj') Sjoberg (1978); open squares with forward slash ('RSp') Rickard and Sjoberg (1983); open square with backward slash ('RSc'), Rickard and Sjoberg (1983); coarsely dashed line ('PWP') with no symbols, Plummer et al. (1978); finely dashed line ('CGW') with no symbols, Chou et al. (1989); open circles ('BP'), Busenberg and Plummer (1986); filled diamond ('MB'), MacInnis and Brantley (1992); inverted filled triangle ('Sc'), Schott et al. (1989); open square with filled lower right diagonal ('ShM'), open square with filled upper right diagonal ('ShS'), both from Shiraki et al. (2000); Solid line with no symbol Dove and Platt (1996); filled triangle ('JR') computed from Jordan and Rammensee (1998); solid square ('VSI'), solid square ('VSIg') "global" rate , Arvidson et al., 2003. Adapted from Geochimica et Cosmochimica Acta, Vol. 67, No. 9, pp. 1623–1634, 2003.

Numerical propagation of the rate uncertainty

We have used a Monte Carlo (MC) scheme to evaluate the impact of uncertainty on the overall distribution of chemical species. Also, a nest MC was established for each k_1'. Two thousands MC simulation have been conducted. Each run takes ~10mins on a single CPU processor. This is considered as a fast simulation given the number of species and the DAE algorithm. The average evolution of Ca^{2+} is given on Figure 9. To assess the impact of the uncertainty, we have also added the upper and the lower limit of the spread due to the uncertainty propagation. Similar behavior has been observed in the remaining species. It is worth noting that the spread of concentration is asymmetric with respect to the mean. This implies that the uniform

distribution of the k_1' leads to non-Gaussian (or at least non-symmetric) distribution of the concentration at every time step.

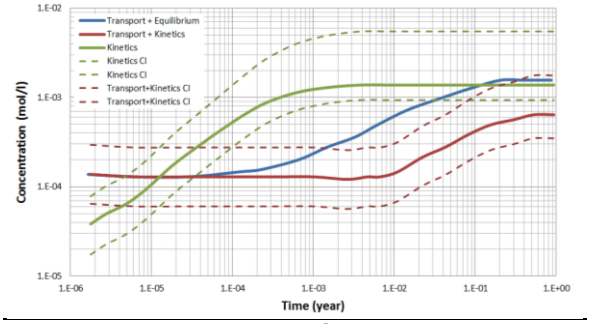


Figure 9: Evolution of Ca^{2+} for different physic-chemical processes with k_1'=15 and confidence interval (CI) when k_1' is iid between [10,20]

CONCLUSIONS

We have built a framework to assess not only the physical and geological uncertainties through the model but also the chemical reactions, especially when dealing with kinetic reactions. The framework has been demonstrated on in one dimensional case and for geochemical dissolution of the calcite. More analyses are on the way to study the coupling effect of the physical and chemical uncertainties on the overall chemical response of the system.

REFERENCES

1. Bryant, R.S. Schechter & L.W. Lake (1986), Interaction of precipitation/dissolution waves and ion exchange in flow trough permeable media. AIChE, May 1986, 32(5):751-764.
2. Cederberg G. A., (1985), A groundwater mass transport and equilibrium chemistry model for multicomponent systems. Water Resour. Res. 21(8):1095-1104.
3. Ezzedine, S., Stochastic modeling of flow and transport in porous and fractured media, Chapter 154 in encyclopedia of hydrological sciences, 34 pages, Wiley 2005.
4. Ezzedine, S., Coupled THMC processes in Geological Media using Stochastic Discrete Fractured Network. Application to HDR Geothermal Reservoirs. Eos Transactions AGU, 89(53), Fall Meeting Supplement, Abstract H41A-084, 2009.
5. Ezzedine, S., I. Lomov, L. Glascoe & T. Antoun, A Comprehensive flow and heat and mass transport UQ in discrete fracture network systems. AGU Fall meeting, San Francisco, Dec 2010.
6. Ezzedine, S., Impact of Geological Characterization Uncertainties on Subsurface Flow Using Stochastic Discrete Fracture Network Models. Annual Stanford Workshop on Geothermal Reservoir Engineering, Stanford, Feb 2010.
7. Ezzedine, S., L. Glascoe, T. Antoun, A Stochastic Framework for UQ and Risk Assessment of EGS. Geothermal Recourses Council Annual Meeting 2010, Sacramento, Oct 2010.
8. Ezzedine, S., Morris, J.P, Effect of Uncertainty of Geomechanical Properties, Fracture Orientations and In-Situ Stress Conditions on Fault Activation. 45th Rock Mechanics Symposium- 2011.
9. Grove D.B. and Wood W.W. (1979), Prediction and field verification of subsurface-water quality changes during artificial recharge, Lubbock, Texas. Groundwater 17(3):250-257.
10. Jennings A.A., Kirkner D.J., Theis I. L., (1982), Multicomponent equilibrium chemistry in groundwater quality models. Water Resour. Res. 18(4):1089-1096.
11. Kinzelbach and Schafer (1989), Coupling of chemistry and transport, in Groundwater Management: Quantity and Quality, Proc. of the Benidorm Symp.,Oct. 1989, pp. 237-259, Int. Ass. of Hydrological Sc., Gentbrugge, Begium, 1989.
12. Kirkner D. J., Theis T.L., Jennings A.A, (1984), Multicomponent solute transport with sorption soluble complexation. Ad. in Water Resour. 7:120-125.
13. Kirkner D. J., Theis T.L., Jennings A.A, (1985), Multicomponent mass transport with chemical interaction kinetics. J. Hydrol. 76:107-117
14. Lewis et al., (1987), Solute transport with equilibrium aqueous complexation and either sorption or ion exchange: Simulation methodology and applications. J. Hydrol. 90:81-115.
15. Miller C.W & L.V. Benson (1983), Simulation of solute transport in a chemically reactive heterogeneous system: Model development and application. Water Resour. Res., 19(2):381-391, April 1983.
16. Morel F., (1983), Principles of aquatic chemistry, John Wiley & Sons, 446p.

17. Nitao, J. J., 1998, Reference manual for the NUFT flow and transport code, version 2.0, Lawrence Livermore National Laboratory, Livermore, CA (UCRLMA-130651).

18. Noorishad C. L.Carnahan L. V.Benson Development of the non-equilibrium reactive chemical transport code CHEMTRNS Rep. LBL-22361 Lawrence Berkeley Lab., Univ. of Calif. Berkeley 1987

19. Plummer L.N., Wigley, T. M. and Parkhurst, D. L., The kinetics of calcite dissolution in CO2-water system at 5 to 60C and 0 to 1 atm CO2. Am. J. Sci., 278, p 179-216, 1978.

20. Reed M. H., (1982), Calculation of multicomponent chemical equilibria and reaction processes in systems involving minerals, gases and an aqueous phase. Geochim. & Cosmoschimica Acta vol.46, pp513-528, 1982

21. Rubin J. and James R.V. (1973), Dispersion-affected transport of reacting solutes in saturated porous media: Galerkin method applied to equilibrium-controlled exchange in unidirectional steady water flow. Water Resour. Res. 9(5):1332-1356.

22. Rubin J. (1983), Transport of reacting solutes in porous media: Relation between mathematical nature of problem formulation and chemical nature reactions. Water Resour. Res. 19(5):1231-1252.

23. Valocchi et al., (1981), transport of ion-exchanging solutes in groundwater: Chromatographic theory and field simulation. Water Resour. Res 17(5):1517-1527.

24. Walsh M.P., Bryant S.L., Lake L.W. (1984), Precipitation and dissolution of solids attending flow trough porous media. AIChE J. 30(2):317-328.

25. Yeh G.T. and V. S. Tripathi, (1989), A critical evaluation of recent developments in Hydrogeochemical transport models of reactive multichemical components. Water Resour. Res. 25(2):93-108, January 1989

26. Yeh G.T. and V. S. Tripathi, (1991), A model for simulating transport of reactive multispecies components: model development and demonstration Water Resour. Res., 27(12):3075-3094, December 1991

27. Xu, T., and K. Pruess. 1998. Coupled modeling of non-isothermal multiphase flow, solute transport and reactive chemistry in porous and fractured media: 1. Model development and validation. Rep. LBNL-42050.

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Material inversion with SW4mopt

Björn Sjögreen[1] N. Anders Petersson[1]

September 27, 2013

[1]Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, PO Box 808, Livermore CA 94551.

Chapter 1

Introduction

SW4mopt is a solver for material inversion and source inversion, built on the forward solver *SW4*. Running *SW4mopt* is very similar to running *SW4*. *SW4mopt* uses the same input commands as *SW4*, and provides an extended set of commands, related to the inverse problem.

Given time series data at a number of stations, *SW4mopt* solves for the material density, ρ, and Lamé parameters μ and λ, in a material that has been parameterized in some way. Currently, the only available parameterization is a representation of the material on a coarse grid. The material at the grid points is defined by interpolation from this coarse grid. *SW4mopt* is written such that it will be easy to extend to other types of parameterizations, e.g., representation as B-splines or a mesh free unstructured representation.

For minimization, *SW4mopt* provides the choice of the limited memory BFGS method (L-BFGS) or a non-linear conjugate gradient method. These are run together with a line search algorithm to determine the step length.

As of writing, September 2013, *SW4mopt* has been successfully run on simple problems with synthetic data, some of them reported below. It is a complete solver for the inverse problem. However, it is far from being ready to be released to the users. *SW4mopt* is not as robust as *SW4*, and we need to test it on many other problems, to make sure that it works as expected, and to gain experience of material inversion. Furthermore, since *SW4mopt* is still in development, it produces a lot of output messages that might be confusing to the average user.

The following should be done to improve *SW4mopt*.

- Implement the material parameterization for grids with topography. Topography is the only critical part that is missing. The material

gradient can be computed with topography, it is just the coarse grid parameterization that has not been done.

- Implement a material parameterization which is distributed on the processors. Currently one copy of the entire coarse material grid is stored in each processor. This will lead to memory limitations as problems size increases.

- Possibility to refine the material grid, so that the inversion to a highly resolved material can be made hierarchally. The same holds for the scaling factors. With a higher material resolution, we can not expect to compute these by forming the Hessian. Can scaling factors be interpolated from a coarse grid inversion ?

- Automatic readjustment of the time step when the material speed increases past the largest stable CFL number. This must now be handled by manual restart, it would be better to have the code doing it automatically.

A few more items that should be addressed, but which require new research and development, and new software.

- Material inversion in an anisotropic material.

- Homogenization techniques to recover isotropic material properties from the result of anisotropic material inversion.

- Material inversion with attenuation.

Chapter 2

The material description

2.1 The mparcart command

The `mparcart` defines a material that is discretized on a coarse Cartesian grid, below refered to as the "material grid". The parameters are the offset values of ρ, μ, and λ relative a reference material, at the points of this grid. The material on the computational grid is defined by trilinear interpolation from the values on the coarser material grid. For example, the density, ρ, at a grid point (i, j, k) in the computational grid is

$$\rho_{i,j,k} = I(\{d^{(\rho)}\}, x_i, y_j, z_k) + \rho^{(0)}_{i,j,k}$$

where $\rho^{(0)}$ is the reference material, and $d^{(\rho)}$ is the difference $\rho - \rho^{(0)}$ on the material grid. The interpolation operator

$$I(\{u\}, x, y, z)$$

evaluates a function u defined at the points of the material grid, at the point (x, y, z).

The number of points in the material grid, and initial values for $d^{(\rho)}, d^{(\mu)}, d^{(\lambda)}$ are specified by the `mparcart` command. For example

```
mparcart nx=5 ny=5 nz=3 init=0
```

defines a material grid with $5 \times 5 \times 3$ points, and with $d^{(\rho)}$, $d^{(\mu)}$, $d^{(\lambda)}$ initialized to zero at all points. The reference material is specified by one of the material commands of *SW4*, e.g., `block` or `pfile`, see the *SW4* User's Guide for a complete description of these material commands.

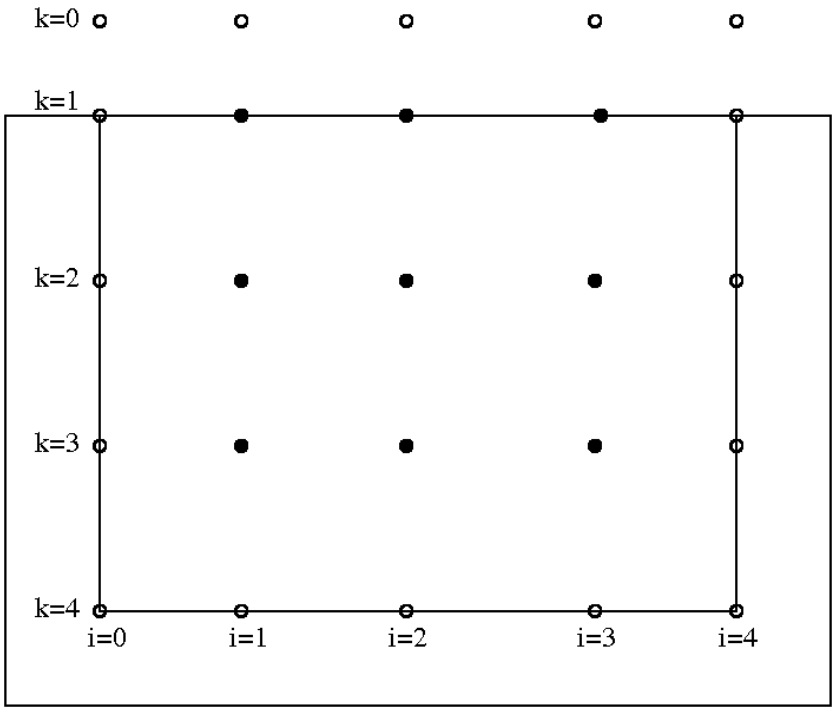


Figure 2.1: Two dimensional slice of a material grid with $n_x = n_y = n_z = 3$. The free surface boundary is at $k = 1$, there are super-grid sponge layers on the other sides. Filled circles indicate unknown parameters, open circles are fixed boundary values.

Using a coarse material grid reduces the number of unknowns, compared with using the material at each grid point as parameters. Furthermore, the resolution in the material is limited in terms of highest frequencies of the computed wave field, making it unreasonable to expect to resolve the material down to the resolution of the computational grid. We recommend that the grid spacing of the material grid is around 10 times the grid spacing of the computational grid. Also to note, the grid spacing of the material grid do not need to have any relation to the spacing of the computational grid. These two grids are independent.

The material grid discretizes only the interior part the domain, without the super-grid sponge layers. The configuration is outlined in Fig. 2.1. The

material grid is given by

$$x_i = x_{min} + ih_x \qquad i = 0, \ldots, n_x + 1 \tag{2.1}$$
$$y_j = y_{min} + jh_y \qquad j = 0, \ldots, n_y + 1 \tag{2.2}$$
$$z_k = z_{min} + kh_z \qquad k = 0, \ldots, n_z + 1 \tag{2.3}$$

where the grid spacings h_x, h_y, and h_z are determined such that $x_0, y_0, x_{n_x+1}, y_{n_y+1}$, and z_{n_z+1} are located at the interface between the interior domain and the super-grid sponge layer. In the depth direction, z_1 is on the free surface and, hence $z_{min} = -h_z$. The point z_0 is above the topography, but it will never be used in the interpolation.

At the first and last points in each direction, ($i = 0$, $j = 0$, $k = 0$, $n_x + 1$, $n_y + 1$, and $n_z + 1$), the offsets $d^{(\rho)}, d^{(\mu)}$, and $d^{(\lambda)}$ are fixed at zero, hence these values are not part of the parameter vector. The total number of unknowns for the material inversion is $3 \times n_x \times n_y \times n_z$.

Chapter 3

Computed examples

3.1 Material grid with a single point

This test problem is defined on a domain of size $35000 \times 35000 \times 17000$. The material is constant with $\rho = 2650$, $v_s = 2437.56$, and $v_p = 4630.76$, giving $\mu = 1.57455 \times 10^{10}$ and $\lambda = 2.5335 \times 10^{10}$. The elastic wave equation is discretized on a grid with spacing $h = 200$. A traction free boundary condition is imposed at the boundary $z = 0$. All other boundaries have super-grid sponge layers of width 2500. Waves are set off by a moment source located at $x_s = y_s = 17500$, $z_s = 2000$. The only non-zero element of the moment tensor is $M_{xy} = 1.0 \times 10^{18}$. The source time function is a Gaussian with $t_0 = 1.5$ and $\omega = 4$.

We consider a material grid of size $1 \times 1 \times 1$, i.e., there is only one point in the material parameterization. The total number of material parameters is 3, i.e., ρ, μ, and λ at the single point. The outline of the material grid in Fig. 2.1 shows that with a single point, it will be located at $x = y = 17500$ and $z = 0$.

Synthetic seismograms are computed at stations on a centered 5×5 grid with spacing 3000 on the surface, ($z = 0$). These seismograms, computed with the constant material, are used as the measured data in the numerical experiments. In these experiments, we initialize the values at the material grid point by a perturbation of the constant material. Denoting the constant, reference, density by $\rho^{(ref)}$, and the perturbation by $d^{(\rho)}$, we have

$$\rho = \rho^{(ref)} + d^{(\rho)}$$

at the material grid point. The notation for μ and λ are similar. The

following five initial perturbations are considered

Name	$d^{(\rho)}$	$d^{(\mu)}$	$d^{(\lambda)}$
pp10	$0.1\rho^{(ref)}$	$0.1\mu^{(ref)}$	$0.1\lambda^{(ref)}$
mm10	$-0.1\rho^{(ref)}$	$-0.1\mu^{(ref)}$	$-0.1\lambda^{(ref)}$
mp10	$-0.1\rho^{(ref)}$	$0.1\mu^{(ref)}$	$0.1\lambda^{(ref)}$
mp30	$-0.3\rho^{(ref)}$	$0.3\mu^{(ref)}$	$0.3\lambda^{(ref)}$
pm30	$0.3\rho^{(ref)}$	$-0.3\mu^{(ref)}$	$-0.3\lambda^{(ref)}$

The minimizing algorithm should converge to the constant material, i.e., $d^{(\rho)}$ should approach zero. The Hessian at the constant material is

$$H = \begin{pmatrix} 2.9371e-03 & -4.9563e-10 & -1.8236e-12 \\ -4.9563e-10 & 8.7013e-17 & 2.4771e-19 \\ -1.8236e-12 & 2.4771e-19 & 3.6478e-20 \end{pmatrix}$$

The condition number of H is 8.571×10^{16}. The scale factors obtained from the diagonal elements of H, and rescaled to have $s_\rho = \rho^{(ref)}$ are,

$$s_\rho = 2650 \quad s_\mu = 1.54 \times 10^{10} \quad s_\lambda = 7.52 \times 10^{11}$$

with these scale factors, the condition number is 107. Note that s_ρ and s_μ are typical sizes of $\rho^{(ref)}$ and $\mu^{(ref)}$ respectively, while s_λ is around 30 times larger than $\lambda^{(ref)}$. As an illustration, Fig. 3.1 shows the density on the plane $x = 17500$ after 1, 2, 3, and 4 iteration with the L-BFGS method. The initial perturbation at the single material grid point gives a hat shaped initial material. The perturbation disappears as the minimizing iterations converges to the constant material used to compute the synthetics. Figure 3.2 shows the convergence histories for the cases in Table 3.1 with 10% perturbation. The convergence from the 30% perturbation are shown in Fig. 3.3.

With the 10% perturbation, it takes 7-10 iterations to converge the misfit down to its final value. The 30% perturbations require 17-20 iterations.

Next, we investigate the convergence of the material properties at the material grid point. Figures 3.4 and 3.5 show the evolution of the relative error, $d^{(\rho)}/\rho^{(ref)}$ for the density and similarly for μ and λ. The perturbation converges to zero, and as shown by the convergence curves, the material has converged in the picture norm after around 10 iterations for the 10% perturbations. With the 30% initial perturbation, around 20 iterations are needed. Figures 3.4 and 3.5 also show that the amplitudes of the perturbations in

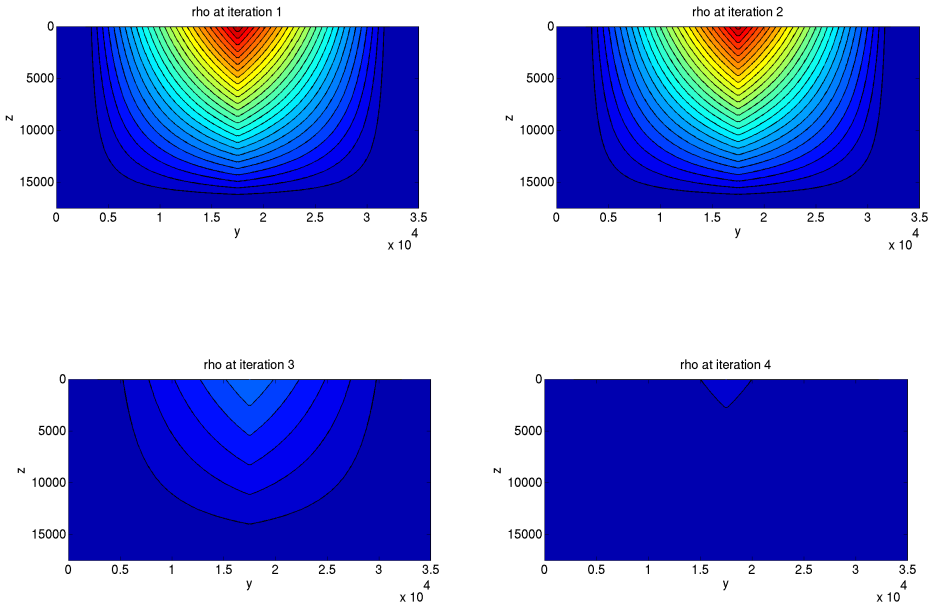


Figure 3.1: Density at $x = 17500$ after iterations 1, 2, 3, and 4. 30 contour levels between 2650 and 2925.

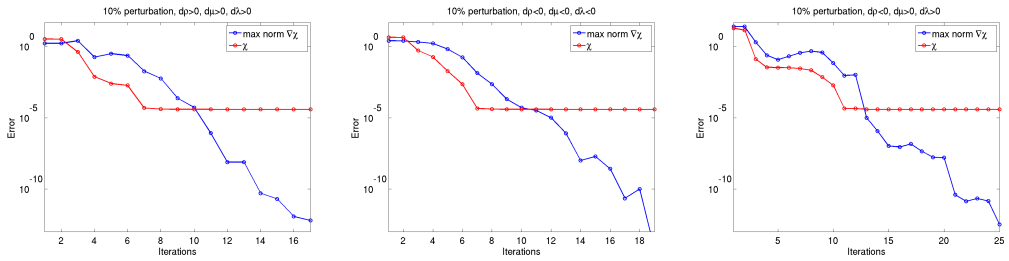


Figure 3.2: Convergence rate of problems pp10 (left), mm10 (middle), and mp10 (right). Blue is the maximum norm of the gradient of the misfit, red is the misfit.

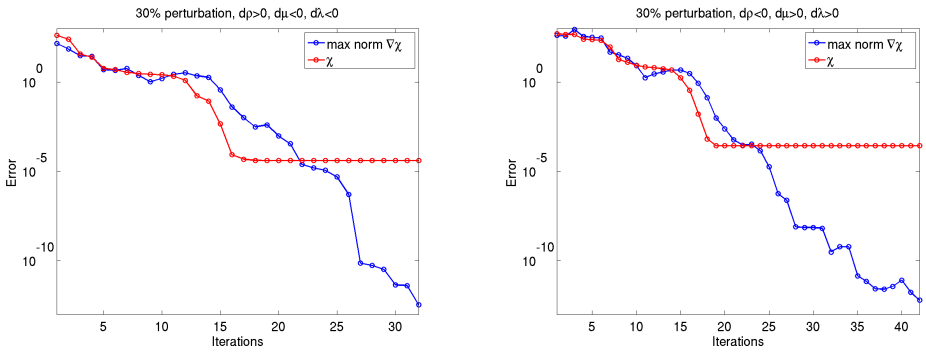


Figure 3.3: Convergence rate of problems pm30 (left), mp30 (right). Blue is the maximum norm of the gradient of the misfit, red is the misfit.

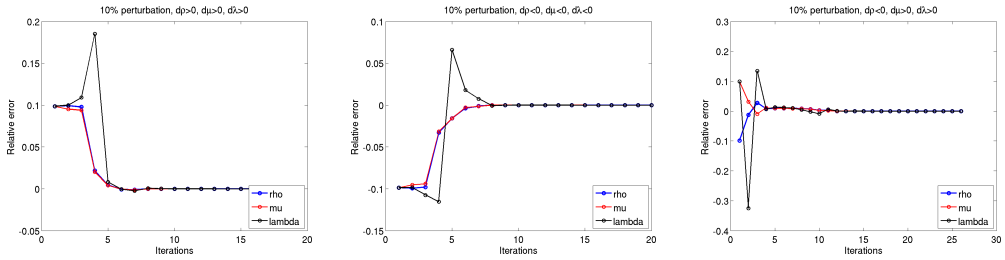


Figure 3.4: Convergence of ρ (blue), μ (red), and λ (black) for problems pp10 (left), mm10 (middle), and mp10 (right).

ρ and μ never exceeds the size of the initial perturbation. λ, on the other hands, have very large variation, especially for the 30% perturbation. This is caused by the scaling factor s_λ being much larger than the size of $\lambda^{(ref)}$. The line search algorithm bases its largest allowed step length on the scale factors, which implies that a much longer relative step is allowed for λ than for ρ and μ.

The input file for this example contains the material specification

```
block vp=4630.76 vs=2437.56 r=2650
mparcart nx=1 ny=1 nz=1 init=onep-pr10.bin
```

meaning that the reference material is defined by the block command, and the initial 10 percent perturbation is read from the file `onep-pr10.bin`. This

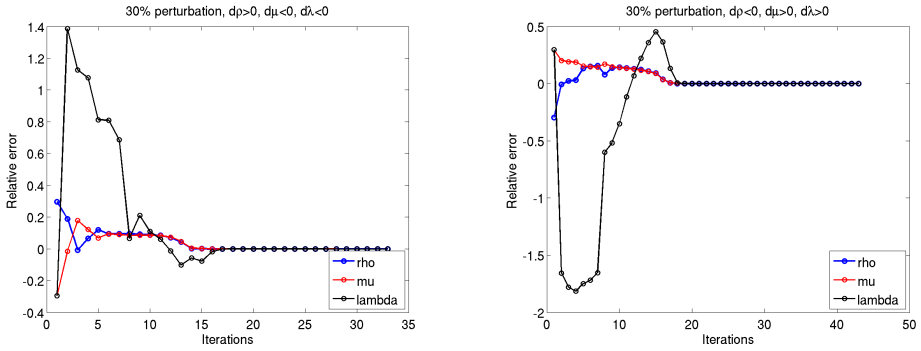


Figure 3.5: Convergence of ρ (blue), μ (red), and λ (black) for problems pm30 (left) and mp30 (right).

file was prepared externally to *SW4mopt*, by a Matlab script. The minimizing algorithm was specified by the lines

```
mrun task=minvert mcheck=on tsoutput=on
lbfgs nvectors=3 maxit=100 tolerance=1e-12 linesearch=on
```

in the inputfile.

3.2 Material with sinusodial variation, $3\times3\times2$ material grid

This test problem has the same dimensions and source as the problem in the previous subsection. The synthetics are computed on a constant material with an added sinusodial variation. The constant material has the properties $\rho = 2650$, $v_s = 2679.5$, and $v_p = 5000$, leading to $\mu = 1.90 \times 10^{10}$ and $\lambda = 2.82 \times 10^{10}$. The amplitude of the sine perturbation is around 10% in ρ, μ and λ. Figure 3.6 shows the density on the plane $z = 1000$ at the exact minimum.

The inversion algorithm starts from the constant material as initial guess, and iterates to find the sinusodial material variation. As an illustration of the convergence process, the density in the plane $z = 1000$, with the same contour levels are plotted in Fig. 3.7.

Just as in the case with one material grid point, the Lamé parameter λ shows the largest variation during the iteration process. The evolution of λ

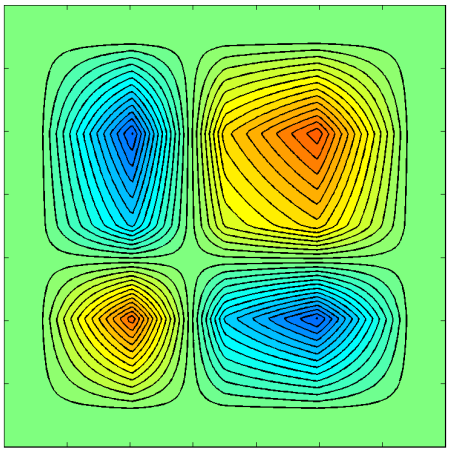


Figure 3.6: Density on the plane $z = 1000$ at the exact minimum.

is displayed in Fig. 3.8, showing it out of range of the plotting contour levels after 10, 20, and 40 iterations.

The convergence in Fig. 3.9, shows that around 200 iterations are needed to drive down the misfit to the level of round-off. However, Figs. 3.7 and 3.8 show that 50–60 iterations give a satisfactory picture of the material, corresponding do a misfit reduction of 4–5 orders of magnitude.

The input file for this example is given by

```
grid h=200 x=35000 y=35000 z=19500
time t=9 utcstart=09/10/2013:23:5:53.000
fileio verbose=1 path=run8 obspath=obs temppath=/tmp/bjorn pfs=1
supergrid gp=13 dc=.015
#
block r=2650 vs=2677.6503357897 vp=4998.11285141419
mparcart nx=3 ny=3 nz=2 init=0
#
mrun task=minvert mcheck=on
lbfgs nvectors=35 maxit=300 tolerance=1e-12 linesearch=on ihess0=scale-factors
mscalefactors rho=2650 mu=1.19e10 lambda=5.10e11
#
```

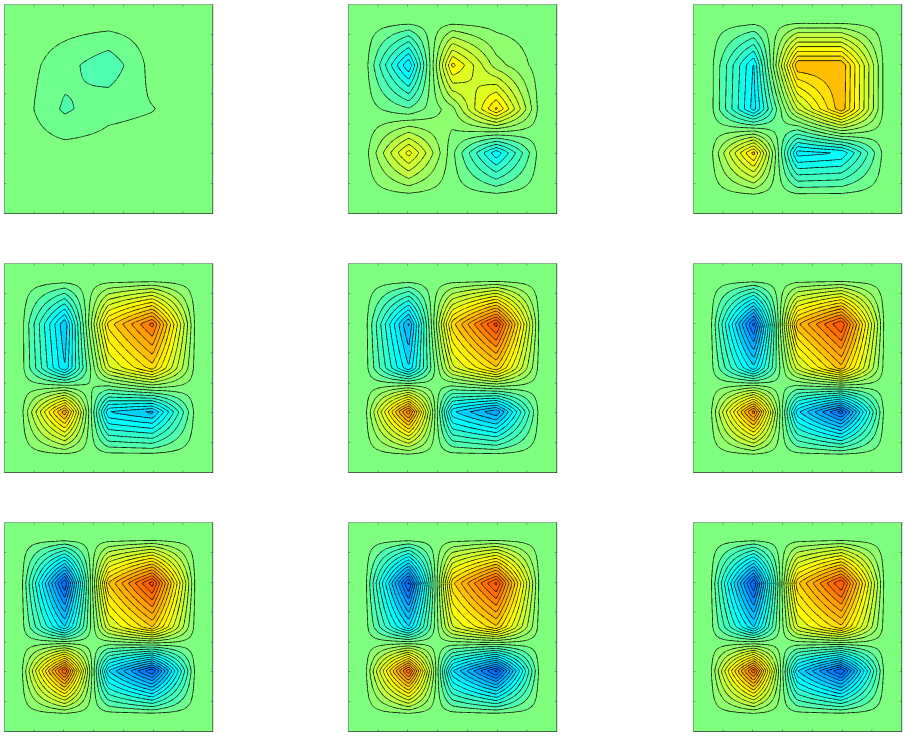


Figure 3.7: Density on the plane $z = 1000$ after 1, 10, 20, 30, 40, 50, 60, 70, and 80 iterations with the L-BFGS method.

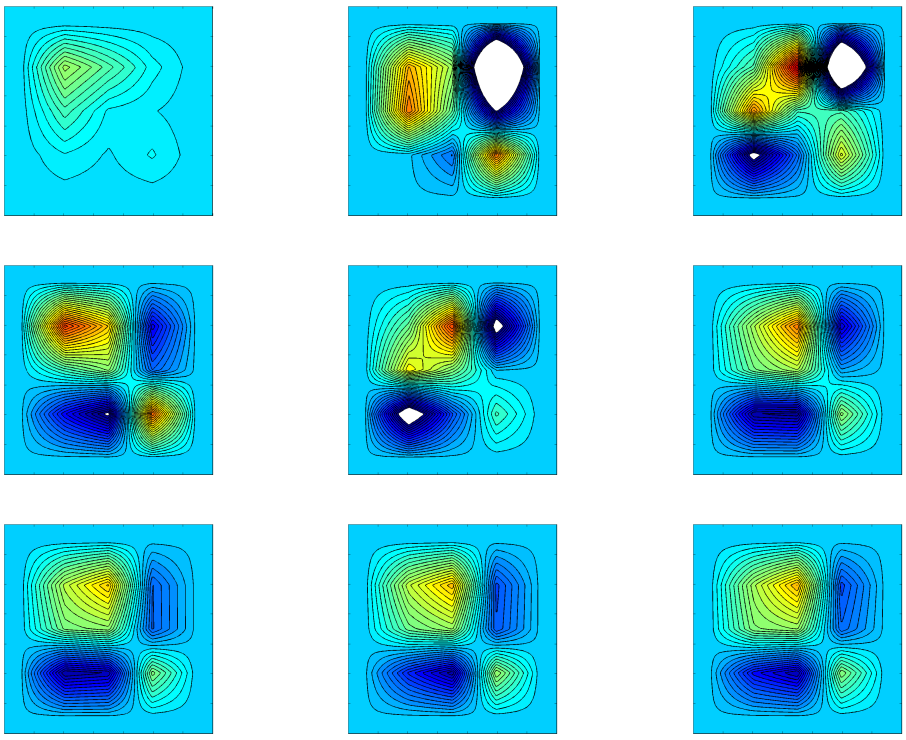


Figure 3.8: Lamé parameter λ on the plane $z = 1000$ after 1, 10, 20, 30, 40, 50, 60, 70, and 80 iterations with the L-BFGS method.

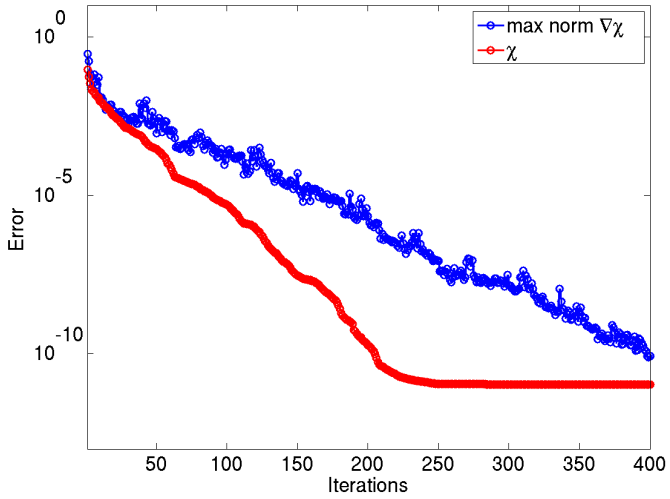


Figure 3.9: Convergence of misfit (red) and maximum norm of the gradient of the misfit (blue) for the computation shown in Fig. 3.7.

```
source x=17500 y=17500 z=2000 Mxy=1 m0=1e18 t0=1.5 freq=4 type=Gaussian
#
developer cfl=1.1
#
observation x=11500 y=11500 z=0 file=sta11
observation x=14500 y=11500 z=0 file=sta21
observation x=17500 y=11500 z=0 file=sta31
 ....
```

The first four lines set up the domain and discretization size. It uses the same syntax as the forward solver, *SW4*. The only additional items for the inverse problem are the two new paths, `obspath` and `temppath`, given at the `fileio` command. The `obspath` is the directory containing the observed seismograms. These data are read by the solver, nothing is output into `obspath`.

The `temppath` is a directory where temporary files associated with the outflow boundaries are stored. In order to reconstruct the forward solution during the backward solve, the forward solution on the outflow boundaries needs to be saved. Each processor that holds a part of the outflow boundary,

writes its own file to `temppath`. These files can become fairly large, and there can be a lot of them. The read and write speeds to `temppath` will have a major effect on the preformance of the inverse solver. Preferably, `temppath` should be a directory on a disk that is local to the computational node. On the LC sytems, the directory `/tmp/username` is usually a good choice, if the files are not too large. The temporary files are deleted by *SW4mopt* when they are no longer needed, however if the program crashes, files might be left in `temppath`.

The next two lines,

```
block r=2650 vs=2677.6503357897 vp=4998.11285141419
mparcart nx=3 ny=3 nz=2 init=0
```

specify the material repesentation and the initial guess material. The computation starts from a zero perturbation on the constant reference material described by the `block` command. The material grid has $3 \times 3 \times 2$ points.

The L-BFGS method is specified by

```
lbfgs nvectors=35 maxit=300 tolerance=1e-12 linesearch=on ihess0=scale-factors
```

A maximum of 300 iterations are taken, using 35 L-BFGS vectors. The initial inverse Hessian will be computed from the scale factors. These are specified on the `mscalefactors` line below. The scale factors are also used to compute the step length in the minimization algorithm.

The source is given below the `mscalefactors` line. The source is specified with exactly the same syntax as in the forward solver. Below the `source` command, the CFL-number is set to 1.1. In *SW4mopt*, the maximum allowed CFL-number is hard coded to 1.3. By computing with a somewhat lower CFL-number, we allow the material wave speeds to increase by (1.3-1.1)/1.1 = 18% during the minimization. *SW4mopt* never changes the initially computed time step. However, the line search algorithm restricts the step size of the minimizer so that the maximum CFL-limit 1.3 is always respected. If the material makes the CFL-number getting too close to 1.3, the minimization step length will go to zero, and the minimizer fails. In that case, the computation has to be restarted with a smaller CFL-number.

The synthetic seismograms were prepared by the forward solver, run in the perturbed material. The seismograms are output in USGS-format, on files `sta11.txt`, `sta21.txt`, etc. up to `sta55.txt`. The synthetics are read into *SW4mopt* by the commands

```
observation x=11500 y=11500 z=0 file=sta11
observation x=14500 y=11500 z=0 file=sta21
observation x=17500 y=11500 z=0 file=sta31
....
```

etc. for all 25 stations. The synthetics were time stamped by the forward solver. The UTC time is registered in the header of the output time series on USGS-format. In this example, the UTC time was September 10th 2013 at 23:5:53. It is important that we specify the same start time for the inverse solver. This is done by the command

```
time t=9 utcstart=09/10/2013:23:5:53.000
```

in the input file.

Chapter 4

Keywords in the input file

The syntax of the input file is the same as in the forward solver, *SW4*. The input file consists of a number of lines with statements

```
command1 parameter1=value1 parameter2=value2 ... parameterN=valueN
# comments are disregarded
command2 parameter1=value1 parameter2=value2 ... parameterM=valueM
...
```

Each command starts at the beginning of the line and ends at the end of the same line. Blank and comment lines are disregarded. A comment is a line starting with a # character. The order of the parameters within each command makes no difference.

Parameter values are either integers (-2,0,5,...), real numbers (20.5, -0.05, 3.4e4), or strings (earthquake, my-favorite-simulation). Note that there must be no spaces around the = signs and strings are given without quotation marks and must not contain spaces.

A breif description of all commands is given in the following sections. The commands marked as [required] must be present in all *SW4mopt* input files, while those marked as [optional] are just that.

4.1 Material optimizer

4.1.1 The mrun command

Syntax:

`mrun task=... mcheck=... tsoutput=...`
Required parameters:
None

The `mrun` command specifies which task to perform. The possible tasks are given in the table below.

possible values of the task option	
Option	**Description**
minvert	Perform material inversion (default)
gradtest	Test gradient computation vs. a numerical derivative
hesstest	Compute the Hessian numerically
func1d	Compute and output a one dimensional cut of the misfit
func2d	Compute and output a two dimensional surface of the misfit
forward	Run one forward solve and exit.
minvert+src11	Perform material and source inversion, 11 parameter source.
minvert+src10	Perform material and source inversion, 10 parameter source.
minvert+src9	Perform material and source inversion, 9 parameter source.
minvert+src6	Perform material and source inversion, 6 parameter source.

Further specification of the `func1d`, `func2d` tasks, e.g., which parameter to vary, can be done with the `msurf` command. The generated cut/surface is output on a file named `fsurf.bin`, the format of which is described in Section **??**.

The intended use of `task=forward` is to generate synthetic seismograms for testing the material inversion.

`mcheck=on` tells the optimizer to check the computed material for reasonableness, e.g., that the density and μ are positive, after each iteration. `tsoutput=on` causes the time series (synthetic seismograms) to be output after each iteration. `mcheck` and `tsoutput` are only effective when the task is `minvert`.

mrun command parameters			
Option	**Description**	**Type**	**Default**
task	task to perform	string	minvert
mcheck	material checking (on or off)	string	off
tsoutput	output time series (on or off)	string	off

Currently, mcheck only outputs diagnostic messages, no attempt to correct the material if it is out of range is made.

4.1.2 The lbfgs command

Syntax:
`lbfgs nvectors=... ihess0=... maxit=... tolerance=... linesearch=...`
Required parameters:
None

Configure the L-BFGS method for minimizing the misfit. L-BFGS will iterate until `maxit` iterations are reached, or until the maximum norm of the scaled gradient of the misfit is less than `tolerance`.

The option `linesearch=off` switches off the line search step of L-BFGS. This is usually not stable. The default `linesearch=on` switches on a standard line search algoritm. However, L-BFGS is only guaranteed to be stable if the so called Wolfe condition is satisfied. The option `linesearch=wolfe` switches on the line search with additional logic to satisfy the Wolfe condition. Line search with the Wolfe condition is computationally expensive, since the gradient of the misfit has to be evaluated at least once. Therefore, it is advisable to try the standard line search first, we have found it to work satisfactory in many cases.

L-BFGS builds and approximation of the inverse Hessian, represented by `nvectors` vectors, which can be thought of as a rank-`nvectors` approximation. The convergence rate is usually better for larger values of `nvectors`. The method requires an initial guess for the inverse Hessian. The option `ihess0=scale-factors` uses an initial guess based on the scale factors of the problem, while `ihess0=gamma` uses an initial guess by an estimate of the size of the Hessian along the search directions, given by formula (7.20) in [**?**].

lbfgs command parameters			
Option	**Description**	**Type**	**Default**
nvectors	Number of l-bfgs vectors to keep	int	10
ihess0	Initial guess for inverse Hessian	string	gamma
maxit	Maximum number of iterations	int	10
tolerance	Termination criterion for gradient	float	10^{-12}
linesearch	Line search method (on, off, or wolfe)	string	on

4.1.3 The nlcg command

Syntax:
`nlcg maxit=... tolerance=... linesearch=... subtype=... maxsubit=...`
Required parameters:
None

Configure the non-linear conjugate gradient (NLCG) method for minimizing the misfit. NLCG will iterate until `maxit` restarts are reached, or until the maximum norm of the scaled gradient of the misfit is less than `tolerance`. CG methods are usually restarted every nth iteration, where n is the number of unknowns. The behavior of the iterations is controlled by the two parameters `maxit` and `maxsubit`. `maxit` gives the maximum number of restarts, and `maxsubit` is the number of iterations between the restarts. By default `maxsubit` is set to n.

The option `linesearch=off` switches off the line search step in NLCG. The default `linesearch=on` switches on a standard line search algoritm. The initial step size is computed by approximating the minimization functional by a quadratic surface.

The `subtype` option makes it possible to specify the Fletcher-Reeves or the Polak-Ribière variants of the NLCG. The Polak-Ribière algorithm forces a restart more often than Fletcher-Reeves. With `subtype=polak-ribiere`, `maxit` needs to be set large enough to allow for the additional restarts.

nlcg command parameters			
Option	**Description**	**Type**	**Default**
maxit	Maximum number of iterations	int	10
tolerance	Termination criterion for gradient	float	10^{-12}
linesearch	Line search method (on or off)	string	on
subtype	fletcher-reeves or polak-ribiere	string	polak-ribiere
maxsubit	Number of subiterations in CG	int	# unknowns

4.1.4 The mscalefactors command

Syntax:
`mscalefactors rho=... mu=... lambda=... file=... misfit=...`
Required parameters:
None

Introducing scale factors will improve the convergence rate of the minmizer, by reducing the condition number of the Hessian of the misfit. For quadratic problems, the scale factors form a diagonal pre-conditioning matrix. There is one scale factor for each unknown. Ideally, the scale factor for the ith unknown, x_i, should be $1/\sqrt{H_{ii}}$, where H_{ii} is the ith diagonal element of the Hessian. The `file=` option specifies a file name, from which the scale factors are read. The number of scale factors on the file should be equal to the number unknown parameters. The format of the file is described in Section **??**.

If the Hessian is not known, an alternative is to set the scale factors to reference sizes of the parameters. If the material parameterization is made such that each parameter is identifiable with one of the material properties, ρ, μ, or λ, the options `rho=`, `mu=`, and `lambda=` can be used to specify three different scales. These scale factors are used throughout for all unknowns of the respective type.

The `misfit=` is a factor that modifies the input scale factors. It is based on the observation that the scale of $1/\sqrt{H_{ii}}$, with $H = \partial^2 f/(\partial x_i^2)$, is actually $x_r/\sqrt{f_r}$, where x_r is the scale of the unknown x_i, and f_r is the scale of the objective function. All input scale factors will be divided by the square root of the value of the `misfit` parameter. The condition number of the Hessian is not affected by multiplying all factors by a constant, but the multiplier will affect the maximum step length, and is currently needed to sometimes

reduce the allowed step size. This is a temporary measure that should be removed, once the line search algorithm has been improved.

mscalefactors command parameters			
Option	**Description**	**Type**	**Default**
rho	Density scale factor	float	1
mu	Scale of Lamé parameter μ	float	1
lambda	Scale of Lamé parameter λ	float	1
file	Name of file containing scale factors	string	None
misfit	Multiplier for scale factors	float	1

4.1.5 The mfsurf command

Syntax:
`mfsurf var=... i=... j=... k=... pmin=... pmax=... npts=... var2=... i2=... j2=... k2=... pmin2=... pmax2=... npts2=...`
Required parameters:
None

The command `mfsurf` controls the selections for `mrun task=func1d` and `mrun task=func2d`. The command assumes that the material parameters can be interpreted as values of ρ, μ, or λ on a logically rectangular grid, for example, when using material parameterization by the `mparcart` command.

`var=`, `i=`, `j=`, `k=` specify one parameter. For example the density at the point with index $(3, 2, 4)$ in the coarse material grid defined by `mparcart`, is selected by

```
mfsurf var=rho i=3 j=2 k=4
```

The command

```
mfsurf var=rho i=3 j=2 k=4 npts=30 pmin=-250 pmax=250
```

specifies the misfit as function of this parameter on the interval [-250,250], discretized by 30 points. To compute and save the specified function on a file, run *SW4mopt* with `mrun task=func1d`.

The second set of input variables, `var2=`, `i2=`, etc. are used for the second dimension when a two dimensional misfit surface is specified with `mrun task=func2d`.

mfsurf command parameters			
Option	**Description**	**Type**	**Default**
var	variable (rho, mu, or lambda)	string	rho
i	i-index	int	1
j	j-index	int	1
k	k-index	int	1
pmin	lower parameter limit	float	-300
pmax	upper parameter limit	float	300
npts	Number of discretization points	int	10
var2	variable (rho, mu, or lambda)	string	rho
i2	i-index	int	1
j2	j-index	int	1
k2	k-index	int	2
pmin2	lower parameter limit	float	-300
pmax2	upper parameter limit	float	300
npts2	Number of discretization points	int	10

4.2 Material parameterization [required]

Several ways to parameterize the material will be tried. Currently, only parameterization through a coarser grid is possible, by the `mparcart` command.

4.2.1 mparcart

Syntax:
`mparcart nx=... ny=... nz=... init=...`
Required parameters:
`nx, ny, nz, init`

The command mparcart defines the material by interpolation from a coarse grid. The coarse grid stores the offsets in ρ, μ and λ from a reference material. The init parameter is either 0, meaning that all offsets are initialized to zero, or the name of a file with previously computed offsets. If a file name is specified, the material is initialized with the values stored on the file. The material optimizer stores the current values of the offsets on the file `parameters.bin` after each iteration. Hence, a previous computation can be restarted by specifying `init=parameters.bin`.

mparcart command parameters			
Option	**Description**	**Type**	**Default**
nx	Number of points in x	int	none
ny	Number of points in y	int	none
nz	Number of points in z	int	none
init	initial guess, file name or 0	string	none

Currently, the complete material grid is stored in each processor. For very large problem sizes, the amount of memory can be a limitation.

4.3 Output

4.3.1 mimage

Syntax:
`mimage x=... y=... z=... cycle=... cycleInterval=...`
`file=... mode=... precision=...`
Required parameters:
Location of the image plane (x, y, or z)
Time for output (cycle, or cycleInterval)

Material images are similar to the image command of *SW4*. The main difference is that `mimage` defines image output related to the iterations of the minimization algorithm. In the `cycle` and `cycleInterval` options, one cycle is interpreted as one iteration of the minimization algorithm. The options `x`, `y`, `z`, `file`, `mode`, and `precision` are identical to the options with the same names in the `image` command. `mimage` only outputs images of material properties. The supported modes are given in the table below.

mimage mode options	
Value	**Description**
rho	Density
lambda	1st Lamé parameter
mu	2nd Lamé parameter (shear modulus)
p	Compressional wave speed
s	Shear wave speed
gradrho	Gradient of misfit w.r.t. density
gradlambda	Gradient of misfit w.r.t. 1st Lamé parameter
gradmu	Gradient of misfit w.r.t. 2nd Lamé parameter
gradp	Gradient of misfit w.r.t. Compressional wave speed
grads	Gradient of misfit w.r.t. Shear wave speed

Note, the misfit gradients are computed with respect to the material properties at each grid point. When using a material parameterization, the misfit with respect to the parameters is given by the chain rule as a combination of these gradients with the derivative of the parameterization.

mimage command parameters			
Option	**Description**	**Type**	**Default**
cycle	Minimizer cycle to output image (≥ 0)	int	0
cycleInterval	Minimizer cycle interval to output a series of images (≥ 1)	int	1
file	File name header of image	string	mimage
precision	Floating point precision for saving data (float or double)	string	float
mode	The field to be saved	string	rho

J Sci Comput
DOI 10.1007/s10915-013-9760-6

Source Estimation by Full Wave Form Inversion

Björn Sjögreen · N. Anders Petersson

Received: 15 August 2012 / Revised: 16 May 2013 / Accepted: 22 July 2013

Abstract We consider the inverse problem of estimating the parameters describing the source in a seismic event, using time-dependent ground motion recordings at a number of receiver stations. The inverse problem is defined in terms of a full waveform misfit functional, where the objective function is the integral over time of the weighted L_2 distance between observed and synthetic ground motions, summed over all receiver stations. The misfit functional is minimized under the constraint that the synthetic ground motion is governed by the elastic wave equation in a heterogeneous isotropic material. The seismic source is modeled as a point moment tensor forcing in the elastic wave equation. The source is described by 11 parameters: the six unique components of the symmetric moment tensor, the three components of the source location, the origin time, and a frequency parameter modeling the duration of the seismic event. The synthetic ground motions are obtained as the solution of a fourth order accurate finite difference approximation of the elastic wave equation in a heterogeneous isotropic material. The discretization satisfies a summation-by-parts (SBP) property that ensures stability of the explicit time-stepping scheme. We use the SBP property to derive the discrete adjoint of the finite difference method, which is used to efficiently compute the gradient of the misfit. A new moment tensor source discretization is derived that is twice continuously differentiable with respect to the source location. The differentiability makes the Hessian of the misfit a continuous function of all source parameters. We compare four different gradient-based approaches for solving the constrained minimization problem; two non-linear conjugate gradient methods (Fletcher–Reeves and Polak–Ribière), and two quasi-Newton methods (BFGS and L-BFGS). Because the Hessian of the misfit has a very large condition number, the parameters must be scaled before the minimization problem can be solved. Comparing several scaling approaches, we find that the diagonal of the Hessian provides the most reliable scaling alternative. Numerical experiments are

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This is contribution LLNL-JRNL-573912.

B. Sjögreen (✉)· N. A. Petersson
Center for Applied Scientific Computing, L-422, LLNL, P.O. Box 808, Livermore, CA 94551, USA
e-mail: sjogreen2@llnl.gov

Published online: 07 August 2013

presented for estimating the source parameters from synthetic ground motions in two different three-dimensional models; one in a simple layer over half-space, and one using a fully heterogeneous material. Good convergence properties are demonstrated in both cases.

1 Introduction

The accurate estimation of seismic source parameters is an important undertaking of modern seismology [15]. Seismic events (e.g. earthquakes, mine collapses, or explosions) generate waves that propagate through the earth and can be recorded by seismometers. These instruments measure the time-dependent ground motion at fixed locations, either on the surface, or in boreholes. Different seismic events generate waves that have different characteristics [1], and the ground motion recordings can be used to estimate the seismic source properties, such as location, source geometry, and source duration. One important application of source estimation is to distinguish between different types of events, i.e., an earthquake, implosion, or explosion. For earthquakes, information about location and geometry (source mechanism) can be used to analyze fault systems and tectonic processes. Source estimation is also becoming increasingly important in monitoring hydraulic fracturing. This technique is used to create fractures in rock such that entrapped hydro-carbons can be released and extracted [16]. A micro-seismic (very small magnitude) event occurs every time the rock fractures. Source estimation can be used to locate these event and in principle also characterize the type of fracture [33].

In this article, we consider seismic source estimation as a minimization problem constrained by the isotropic elastic wave equation subject to appropriate boundary and initial conditions. The objective is to find the source parameters that minimize the distance between the time-dependent recorded and synthetic wave forms. This general approach is known as full waveform inversion (FWI). The technique was first introduced by Lailly [19] and Tarantola [35]. Since then FWI has become an important tool in global seismology, where recordings of long period waves from larger earthquakes can be used to infer large scale material properties of the earth [11,37,38]. Kim et al. [15] used FWI to estimate the moment tensor and source depth from earthquake ground motion observations. FWI can also be used to build material models for geophysical exploration applications, see Virieux and Operto [40] and the references therein.

To introduce FWI for source estimation, we assume that the motion of the ground is observed at the fixed spatial locations $\mathbf{x}_r$, $r = 1, \ldots, R$, and that three orthogonal components of the displacement are measured as functions of time at all recording stations. We denote the measurements by $\mathbf{d}_r(t)$. Let $\mathbf{u}(\mathbf{x}, t)$ be the synthetic displacement field governed by the elastic wave equation. The synthetic displacement depends implicitly on the source parameters, which we collect in the P-dimensional real-valued vector $\mathbf{p}$. Various misfit functionals have been proposed, see Tromp et al. [38] and Tape et al. [34]. Here we define the continuous minimization problem through the misfit functional

$$\mathcal{X}_c(\mathbf{p}) = \frac{1}{2} \sum_{r=1}^{R} \int_{t=0}^{T} s(t) \, |\mathbf{u}(\mathbf{x}_r, t) - \mathbf{d}_r(t)|^2 \, dt, \tag{1}$$

where $s(t) \geq 0$ is a weight function and $|\mathbf{w}|$ denotes the magnitude of the vector $\mathbf{w} \in \Re^3$. Note that the misfit is a non-negative real scalar functional of $\mathbf{u}$, which measures the distance between the time-dependent wave forms $\mathbf{u}(\mathbf{x}_r, t)$ and $\mathbf{d}_r(t)$ in the time interval $0 \leq t \leq T$.

Hence, $\mathcal{X}_c = 0$ implies perfect agreement between the wave forms at all recording stations, i.e., $\mathbf{u}(\mathbf{x}_r, t) = \mathbf{d}_r(t)$ for $0 \le t \le T$ and $r = 1, 2, \ldots, R$.

We focus on smaller seismic events (magnitude $M_w < 5$), where the source can be modeled as a point moment tensor forcing in the elastic wave equation,

$$\mathbf{f}(\mathbf{x}, t; \mathbf{p}) = g(t)\mathcal{M}\nabla\delta(\mathbf{x} - \mathbf{x}_s). \tag{2}$$

Here, $\nabla\delta(\mathbf{x})$ is the gradient of the Dirac distribution. The source time function $g(t; t_0, \omega_0) = \tilde{g}(t - t_0, \omega_0)$ is assumed to depend on two parameters; a time shift t_0 and a frequency parameter ω_0 that controls the duration of the event. The source is located at $\mathbf{x}_s = (x_s,\ y_s,\ z_s)$ and the elements of the symmetric matrix $\mathcal{M}$ are denoted

$$\mathcal{M} = \begin{pmatrix} m_{xx} & m_{xy} & m_{xz} \\ m_{xy} & m_{yy} & m_{yz} \\ m_{xz} & m_{yz} & m_{zz}. \end{pmatrix}$$

Under these assumptions the forcing function $\mathbf{f}$, and the synthetic displacement $\mathbf{u}$, depend on $P = 11$ parameters,

$$\mathbf{p} = (x_s,\ y_s,\ z_s,\ m_{xx},\ m_{xy},\ m_{xz},\ m_{yy},\ m_{yz},\ m_{zz},\ t_0,\ \omega_0). \tag{3}$$

Source estimation by minimizing the full waveform misfit (1) can give unreliable results if the actual material properties are poorly represented by the material model. For example, if the wave speed between the source and one receiver is too high, the synthetic signal will arrive too early at this receiver, making it difficult to match the observed wave form. Further complications occur if the isotropic elastic wave equation does not govern the observed seismic ground motion, for example due to non-linear, anisotropic, or visco-elastic soil behaviors. However, these topics are beyond the scope of the present paper. In the following we assume that the material model is sufficiently accurate and only invert for the source parameters. We remark that some errors in the material model can be masked out by the window function $s(t)$ in the misfit functional (1). This approach can be generalized to use different window functions for each component and receiver [15]. The observations can also be shifted to compensate for arrival time errors [22].

Many different optimization methods could be used for minimizing the misfit functional. For example, Newton-type methods that use both the gradient and the Hessian, or direct search methods that require neither gradient, nor Hessian information. Our motivation for not using a Newton method is that the Hessian is too expensive to evaluate in every iteration. Direct search methods are very robust, but can be computationally inefficient unless the number of parameters is small, see Kolda et al. [17] for a review. Optimization methods that use the gradient, but not the Hessian, provide a computationally efficient alternative. In particular, we consider quasi-Newton methods and nonlinear conjugate gradient methods (NLCG). Quasi-Newton methods use the gradient and an approximate Hessian. These methods can be made computationally efficient by approximating the Hessian by the positive definite secant update, see Dennis and Schnabel [7] or Nocedal and Wright [25] for details. Here we consider the BFGS and the L-BFGS methods. The latter is a limited storage variant of BFGS, where only a small part of the approximate Hessian is retained. This approach is particularly suitable for problems with a large number of parameters, where the full Hessian is impractical or impossible to store. Another interesting class of optimization methods are the non-linear conjugate gradient methods [23]. In this paper we consider the Fletcher–Reeves and the Polak–Ribière variants.

It has been known for a long time that the gradient of a functional such as (1), which is constrained by a partial differential equation (PDE), can be computed by solving the adjoint

PDE. The method of Lagrange multipliers can be used to derive the adjoint PDE, which often is discretized by the same method as the original PDE. Early examples of the adjoint approach include Lions [21], Pironneau [29], and Jameson [13]. Some applications of the adjoint technique to problems from geophysics are given in Plessix [30].

To allow for general heterogeneous elastic materials, we discretize the elastic wave equation by a fourth order accurate finite difference method and solve it numerically. We remark that our discretization is fundamentally different from the popular staggered grid method [12,20,39], which discretizes the elastic wave equation in first order velocity-stress formulation. Our method uses a node based discretization of the elastic wave equation in second order displacement formulation and is fourth order accurate both in space and time. It satisfies the principle of summation by parts (SBP) for heterogeneous materials on Cartesian grids, see Sjögreen and Petersson [31]. This property ensures that the method is energy stable and that the discretization can be generalized to curvilinear coordinates [3]. Hence, it is possible to account for realistic topography by constructing a boundary conforming curvilinear grid. The ability to accurately model free surfaces on realistic (non-planar) topography makes our finite difference method a very attractive alternative to the recently developed finite element [5], spectral element [18], discontinuous Galerkin [8,14], and finite volume [9] discretizations on unstructured grids.

In our approach, we first discretize the elastic wave equation and the misfit functional, and then optimize the source parameters to minimize the discrete misfit. Numerical artifacts due to truncation errors can therefore only enter through the discretization of the elastic wave equation. In particular, there are no additional numerical errors due to the discretization of the adjoint PDE, which may otherwise be the case if the continuous misfit functional is minimized and the elastic wave equation and its adjoint are discretized independently. Because the curvilinear formulation leads to lengthy algebraic formulas that can obscure the presentation, we only describe the source estimation technique for the case of a flat free surface. The generalization to the curvilinear case is straightforward, but rather technical.

Our first main result is presented in Sect. 2. Here we use the SBP property to derive the adjoint of the discretized elastic wave equation and we prove the adjoint relation between the solutions of the discretized elastic and adjoint wave equations and their forcing functions. This relation is used in Sect. 3 to derive an efficient approach for computing the gradient and Hessian of the discrete misfit. Note that a gradient-based minimization method is only guaranteed to converge if the Hessian is a continuous function of all parameters. For this reason the synthetic displacement must be twice continuously differentiable with respect to all parameters. This implies that the forcing in the discrete elastic wave equation must have the same continuity. The forcing term (2) is singular in space and is discretized by imposing a number of moment conditions, which guarantee the accuracy of the solution away from the singularity [26]. The discretization results in a grid function where the coefficients depend on the source location. Our second main contribution is presented in Sect. 4, where we develop a new spatial discretization of the singular source function (2). This source discretization is designed to be compatible with a fourth order accurate difference scheme, and to be twice continuously differentiable with respect to the source location.

The sizes of the parameters in the source estimation problem span many orders of magnitude. In SI-units, $\mathbf{x}_s$ is of the order $\mathcal{O}(10^4)$, the moment tensor components m_{ij} are of the order $\mathcal{O}(10^{15}) - \mathcal{O}(10^{18})$. The start time t_0 and the frequency parameter ω_0 are both between $\mathcal{O}(1)$ and $\mathcal{O}(10)$. Because there is such a large difference in size between the smallest and largest parameter values, the minimization problem is very poorly scaled and the condition number of the Hessian is very large. Scaling is important for both the quasi-Newton and the NLCG methods. For the quasi-Newton methods, the scaling provides an approach for

calculating the initial approximate Hessian. The NLCG methods converge optimally when the scaled Hessian has a condition number of one. Our third main contribution is presented in Sect. 6, where we perform numerical experiments on synthetic problems. We investigate how different scaling strategies affect the condition number of the scaled Hessian, and the convergence rate of the minimization algorithm. We demonstrate that a scaling based on the diagonal of the Hessian provides the most robust alternative of the options considered here.

The remainder of this article is organized in the following way. Section 2 gives an overview of our fourth order accurate finite difference discretization, defines the discrete source estimation problem, derives the adjoint of the discretized elastic wave equation, and proves the adjoint property. Section 3 describes how the adjoint property can be used to compute the gradient and Hessian of the discrete misfit. The spatial discretization of the singular forcing function (2) is described in Sect. 4. Section 5 discusses how an initial guess for the source parameters can be obtained. We perform numerical experiments with the complete source inversion algorithm on synthetic test problems in Sect. 6. Conclusions are given in Sect. 7.

2 The Discretized Problem

In the following we assume that the displacement field $\mathbf{u}(\mathbf{x}, t)$ satisfies the elastic wave equation in the three-dimensional domain Ω, subject to initial and boundary conditions. Here, the boundary is denoted $\Gamma = \Gamma_1 \cup \Gamma_2$. The displacement is governed by

$$\begin{aligned} \rho \mathbf{u}_{tt} &= \nabla \cdot \boldsymbol{\tau}(\mathbf{u}) + \mathbf{f}(\mathbf{x}, t; \mathbf{p}), && \mathbf{x} \in \Omega,\ 0 \le t \le T, \\ \mathbf{u}(\mathbf{x}, 0) &= 0, && \mathbf{x} \in \Omega,\ t = 0, \\ \mathbf{u}_t(\mathbf{x}, 0) &= 0, && \mathbf{x} \in \Omega,\ t = 0, \\ \mathbf{n} \cdot \boldsymbol{\tau}(\mathbf{u}) &= 0, && \mathbf{x} \in \Gamma_1,\ 0 \le t \le T, \\ \mathbf{u} &= 0, && \mathbf{x} \in \Gamma_2,\ 0 \le t \le T, \end{aligned} \tag{4}$$

where ρ is the density and $\mathbf{f}$ is the moment tensor forcing (2). We further assume that the earth can be described as a heterogeneous isotropic elastic material. The stress tensor $\boldsymbol{\tau}(\mathbf{u})$ is then related to the displacement gradient through

$$\boldsymbol{\tau}(\mathbf{u}) = \lambda \, \mathrm{div}(\mathbf{u}) \, I + \mu \, (\nabla \mathbf{u} + \nabla \mathbf{u}^T), \tag{5}$$

where $\lambda(\mathbf{x})$ and $\mu(\mathbf{x})$ are the first and second Lamé parameters of the material.

2.1 A Self-Adjoint Fourth Order Accurate Finite Difference Scheme

Consider the elastic wave Eq. (4) on the box shaped domain $(x, y, z) \in [0, x_{max}] \times [0, y_{max}] \times [0, z_{max}]$, and the time interval $0 \le t \le T$. Let the computational grid be

$$x_i = (i-1)h, \qquad y_j = (j-1)h, \quad \text{and} \quad z_k = (k-1)h,$$

where $h > 0$ is the grid size and i, j, and k are integers. The domain sizes are chosen such that $x_{N_x} = x_{max}$, $y_{N_y} = y_{max}$, and $z_{N_z} = z_{max}$. We use ghost points outside the domain to impose the boundary conditions. Time is discretized on the grid $t_n = n\Delta_t$, where $\Delta_t > 0$ is the fixed time step and n is an integer. The time step is chosen to satisfy the CFL stability condition, and $t_M = M\Delta_t = T$ where $M > 0$ is the total number of time steps.

The numerical approximation of the displacement vector $\mathbf{u}(\mathbf{x}, t)$ at grid point (i, j, k) and time level t_n is denoted by $\mathbf{u}^n_{i,j,k} = (u^n_{i,j,k}, v^n_{i,j,k}, w^n_{i,j,k})$. To improve readability, we occasionally suppress the subscript or superscript on $\mathbf{u}$, for example by writing $\mathbf{u}^n$ for $\mathbf{u}^n_{i,j,k}$.

When convenient we also use the vector index notation $\mathbf{i} = (i, j, k)$ to indicate a spatial grid point index.

In Sjögreen and Petersson [31], we developed a fourth order accurate symmetric discretization of the divergence of the stress tensor (5). This operator, denoted by $\mathbf{L}_h(\mathbf{u})$, has the property that

$$(\mathbf{v}, \mathbf{L}_h(\mathbf{u}))_h = (\mathbf{L}_h(\mathbf{v}), \mathbf{u})_h, \tag{6}$$

for any two grid functions $\mathbf{u}$ and $\mathbf{v}$ that satisfy the discretized boundary conditions

$$\mathbf{B}(\mathbf{u})_{i,j,k} = 0, \quad \mathbf{x}_{i,j,k} \in \Gamma. \tag{7}$$

The scalar product in (6) is defined by

$$(\mathbf{v}, \mathbf{u})_h = h^3 \sum_{k=1}^{N_z} \sum_{j=1}^{N_y} \sum_{i=1}^{N_x} a_{i,j,k} \langle \mathbf{v}_{i,j,k}, \mathbf{u}_{i,j,k} \rangle, \tag{8}$$

where $a_{i,j,k}$ are positive weights determined from the summation by parts property of $\mathbf{L}_h(\mathbf{u})$ that is needed to enforce (6). Also, $\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{q=1}^{3} u^{(q)} v^{(q)}$, is the inner product between real-valued vectors with three components. Using this notation, the magnitude of $\mathbf{u}$ satisfies $|\mathbf{u}|^2 = \langle \mathbf{u}, \mathbf{u} \rangle$.

We consider boundary operators $\mathbf{B}$ that either discretize free surface or Dirichlet boundary conditions,

$$\mathbf{B}(\mathbf{u}^n)_{i,j,k} = \begin{cases} \mathcal{B}(\mathbf{u}^n)_{i,j,k} \mathbf{n}_{i,j,k}, & \text{Free surface,} \\ \mathbf{u}^n_{i,j,k}, & \text{Dirichlet.} \end{cases}$$

Here, $\mathcal{B}(\mathbf{u})$ is a special difference approximation of the stress tensor on the boundary that matches $\mathbf{L}_h(\mathbf{u})_{i,j,k}$ such that (6) is satisfied. The vector $\mathbf{n}_{i,j,k}$ is the outward boundary normal. A detailed description of the interior and boundary discretizations can be found in [24,31].

We discretize the elastic wave Eq. (4) using the fourth order accurate difference method described in Sjögreen and Petersson [31]. This method computes the displacement field $\mathbf{u}^n$, $n = 1, 2, \ldots, M$, starting from initial data $\mathbf{u}^{-1}$ and $\mathbf{u}^0$, as is outlined in Algorithm 1.

Algorithm 1 *4th order accurate predictor–corrector scheme for the elastic wave equation.*

procedure FORWARD($\mathbf{u}$, $\mathbf{F}$)
  Initial conditions: $\mathbf{u}^0 = \mathbf{0}$ and $\mathbf{u}^{-1} = \mathbf{0}$
  for $n = 0, 1, \ldots, M-1$ **do**
    Predictor step:

$$\mathbf{u}^* = 2\mathbf{u}^n - \mathbf{u}^{n-1} + \frac{\Delta_t^2}{\rho} \left(\mathbf{L}_h(\mathbf{u}^n) + \mathbf{F}(t_n; \mathbf{p}) \right)$$

    Impose boundary condition (7) on $\mathbf{u}^*$ to define its ghost point values
    Acceleration: $\mathbf{v}^n = \left(\mathbf{u}^* - 2\mathbf{u}^n + \mathbf{u}^{n-1} \right) / \Delta_t^2$
    Corrector step:

$$\mathbf{u}^{n+1} = \mathbf{u}^* + \frac{\Delta_t^4}{12\rho} \left(\mathbf{L}_h(\mathbf{v}^n) + \mathbf{F}_{tt}(t_n; \mathbf{p}) \right) + \frac{1}{\rho} \mathbf{S}_G(\mathbf{u}^n - \mathbf{u}^{n-1})$$

    Impose boundary condition (7) on $\mathbf{u}^{n+1}$ to define its ghost point values
  end for
end procedure

Note that the grid function $\mathbf{F}(t;\mathbf{p})$ in this algorithm represents a discretization of the singular source term $\mathbf{f}(\mathbf{x},t;\mathbf{p})$ in the elastic wave Eq. (4). This discretization will be described in detail in Sect. 4.

We use the super-grid modeling method [2,27] for reducing reflections from far-field boundaries. We prefer this approach over the well-known perfectly matched layer (PML) technique [4]. Even though the PML technique has been very successful for Maxwell's equations and the acoustic wave equation, it is difficult to design a robust PML for the time-dependent elastic wave equation. Here, heterogeneous material properties and free surface boundary conditions can cause instabilities. In fact, a negative stability result was established by Skelton et al. [32], who showed that growing, back propagating, modes can exist in the PML for a layered elastic material, thereby making the PML equations ill-posed.

In the super-grid method, a coordinate transformation is used to map a very large physical domain to a significantly smaller computational domain, where the elastic wave equation is solved numerically on a regular grid. To damp out waves that become poorly resolved because of the coordinate transformation, a high order dissipation operator is added in layers near the boundaries of the computational domain. In our implementation, the dissipation operator (denoted by $\mathbf{S}_G(\mathbf{u})$ in Algorithm 1) is consistent with

$$-\gamma h^4 \Delta_t \left(\phi^{(x)}(x)(\rho\sigma^{(x)}(x)\mathbf{u}_{xxt})_{xx} + \phi^{(y)}(y)(\rho\sigma^{(y)}(y)\mathbf{u}_{yyt})_{yy} + \phi^{(z)}(z)(\rho\sigma^{(z)}(z)\mathbf{u}_{zzt})_{zz} \right).$$

The coordinate transformation enters through the one-dimensional positive grid stretching functions $\phi^{(x)}$, $\phi^{(y)}$, and $\phi^{(z)}$. These functions are equal to one in the interior of the computational domain, and decrease monotonically to a small value, $0 < \epsilon_L \ll 1$, at the far-field boundaries. The one-dimensional, non-negative, taper functions $\sigma^{(x)}$, $\sigma^{(y)}$, and $\sigma^{(z)}$, control the local strength of the damping. They are zero in the interior of the domain and increase monotonically to unity at the far-field boundaries. The constant $\gamma > 0$ controls the overall strength of damping. The computational domain is terminated by homogeneous Dirichlet boundary condition at the far-field boundaries, such that the dissipation operator satisfies the SBP relation

$$(\mathbf{v}, \mathbf{S}_G(\mathbf{u}))_h = (\mathbf{S}_G(\mathbf{v}), \mathbf{u})_h, \tag{9}$$

for all grid functions that satisfy the boundary conditions. One can prove by energy estimates that the super-grid technique leads to a stable numerical method with decreasing energy. The proof holds for heterogeneous material properties and a free surface boundary condition on one side of the domain. Numerical experiments show that the artificial reflections can be made extremely small by making the super-grid layers sufficiently wide, see Petersson and Sjögreen [27] for details.

2.2 The Discrete Source Estimation Problem

A straightforward generalization of the continuous formula (1) leads to the discrete misfit functional

$$\mathcal{X}(\mathbf{p}) = \frac{1}{2} \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) |\mathbf{u}_{\mathbf{i}_r}^n - \mathbf{d}_r(t_n)|^2. \tag{10}$$

As in the continuous problem, $s(t_n) \geq 0$ is a weight function. We assume that all recording stations coincide with grid points, i.e., $\mathbf{x}_r = \mathbf{x}_{\mathbf{i}_r}$ for some vector index $\mathbf{i}_r = (i_r, j_r, k_r)$. Furthermore, the observed displacements $\mathbf{d}_r(t)$ are assumed to have already been filtered in time such that they only contain motions that can be captured on the computational grid.

Similar to the continuous case, the displacements at the recording stations depend implicitly on the parameter vector $\mathbf{p}$ in the discretized forcing function $\mathbf{F}$. Given the source parameters $\mathbf{p}$, we can use Algorithm 1 to calculate the solution of the elastic wave equation, which then can be inserted into (10) to evaluate the discrete misfit $\mathcal{X}(\mathbf{p})$. Hence, the discrete source estimation problem can be stated as the constrained minimization problem,

$$\min \mathcal{X}(\mathbf{p}), \quad \mathbf{u}^n \text{ is calculated by Algorithm 1 with forcing } \mathbf{F}(t_n; \mathbf{p}).$$

2.3 The Adjoint Wave Equation

An efficient approach for computing the gradient of the misfit uses the adjoint wave field, $\boldsymbol{\kappa}_{\mathbf{i}}^n$. The adjoint wave field satisfies the adjoint of the discretized elastic wave equation. Let the adjoint equation have the source term $\mathbf{G}(t_n)$. A method for calculating $\boldsymbol{\kappa}$, starting from terminal data $\boldsymbol{\kappa}^M$ and $\boldsymbol{\kappa}^{M-1}$, and stepping backward in time is outlined in Algorithm 2.

Algorithm 2 *The adjoint of the 4th order scheme for the elastic wave equation.*

1: **procedure** ADJOINT($\boldsymbol{\kappa}$, $\mathbf{G}$)
2: Terminal conditions: $\boldsymbol{\kappa}^{M-1} = \mathbf{0}$ and $\boldsymbol{\kappa}^M = \mathbf{0}$
3: **for** $n = M-1, M-2, \ldots, 1$ **do**
4: Predictor step:

$$\boldsymbol{\kappa}^* = 2\boldsymbol{\kappa}^n - \boldsymbol{\kappa}^{n+1} + \Delta_t^2 \frac{\mathbf{L}_h(\boldsymbol{\kappa}^n)}{\rho} \tag{11}$$

5: Impose boundary condition (7) on $\boldsymbol{\kappa}^*$ to define its ghost point values
6: Compute acceleration: $\boldsymbol{\zeta}^n = \left(\boldsymbol{\kappa}^* - 2\boldsymbol{\kappa}^n + \boldsymbol{\kappa}^{n+1}\right)/\Delta_t^2$
7: Corrector step:

$$\boldsymbol{\kappa}^{n-1} = \boldsymbol{\kappa}^* + \frac{\Delta_t^4}{12}\frac{\mathbf{L}_h(\boldsymbol{\zeta}^n)}{\rho} + \frac{\Delta_t^2}{\rho}\mathbf{G}(t_n) - \frac{1}{\rho}\mathbf{S}_G(\boldsymbol{\kappa}^{n+1} - \boldsymbol{\kappa}^n), \tag{12}$$

8: Impose boundary condition (7) on $\boldsymbol{\kappa}^{n-1}$ to define its ghost point values
9: **end for**
10: **end procedure**

The adjoint property is made precise in the following theorem.

Theorem 1 *Let* $\mathbf{F}$ *(with second time derivative* $\mathbf{F}_{tt}$*) be the source term in the discretized elastic wave equation, and use Algorithm 1 to calculate* $\mathbf{u}$*. Furthermore, let* $\mathbf{G}$ *be the source term in the adjoint wave equation and use Algorithm 2 to calculate* $\boldsymbol{\kappa}$ *(with acceleration* $\boldsymbol{\zeta}$*). Then the grid functions* $\mathbf{u}$ *and* $\boldsymbol{\kappa}$ *are adjoint in the sense that*

$$\sum_{n=0}^{M-1} \left(\mathbf{G}^n, \mathbf{u}^n\right)_h = \sum_{n=0}^{M-1} \left(\boldsymbol{\kappa}^n, \mathbf{F}^n + \frac{\Delta_t^2}{12}\mathbf{F}_{tt}^n\right)_h + \frac{\Delta_t^2}{12}\sum_{n=0}^{M-1}\left(\boldsymbol{\zeta}^n, \mathbf{F}^n\right)_h. \tag{13}$$

Proof See "Appendix 1".

3 Minimizing the Misfit

We consider two different classes of gradient-based methods for minimizing the discrete misfit; quasi-Newton methods and non-linear conjugate gradient (NLCG) methods.

A NLCG method generalizes the conjugate gradient method to non-quadratic problems [23]. In this paper, we consider the Fletcher–Reeves and Polak–Ribière variants. Preconditioning can be used to improve the convergence properties of NLCG methods and is often necessary to make these methods practically useful. The preconditioning corresponds to the change of variables, $\hat{\mathbf{p}} = S\mathbf{p}$, where S is a non-singular matrix. To introduce the preconditioner, we first formulate the minimization algorithm in the scaled variables, and then transform it back to the original variables. Algorithm 3 shows the preconditioned Fletcher–Reeves NLCG method with m restarts, and where the parameter vector $\mathbf{p}$ has P components. Note that Fletcher–Reeves is restarted every Pth iteration.

Algorithm 3 *The preconditioned Fletcher–Reeves algorithm. Here,* $\nabla\mathcal{X}_k = \nabla\mathcal{X}(\mathbf{p}_k)$.

1: **procedure** PRECOND- FLETCHER- REEVES($\mathbf{p}_0$)
2: **for** $r = 1, 2, \ldots, m$ **do**
3: Initial search direction: $\mathbf{q}_0 = -(S^T S)^{-1}\nabla\mathcal{X}(\mathbf{p}_0)$
4: **for** $k = 0, 1, \ldots, P-1$ **do**
5: Line search: find steplength α_k that minimizes $\mathcal{X}(\mathbf{p}_k + \alpha_k\mathbf{q}_k)$
6: Next parameter vector: $\mathbf{p}_{k+1} = \mathbf{p}_k + \alpha_k\mathbf{q}_k$
7: Compute β_k: (Polak–Ribière: use (14))

$$\beta_k = \frac{\nabla\mathcal{X}_{k+1}^T (S^T S)^{-1}\nabla\mathcal{X}_{k+1}}{\nabla\mathcal{X}_k^T (S^T S)^{-1}\nabla\mathcal{X}_k}$$

8: Polak–Ribière: if $\beta_k \leq 0$, set $\mathbf{p}_0 = \mathbf{p}_{k+1}$ and goto 3
9: Next search direction: $\mathbf{q}_{k+1} = -(S^T S)^{-1}\nabla\mathcal{X}(\mathbf{p}_{k+1}) + \beta_k\mathbf{q}_k$
10: **if** $||S^{-1}\nabla\mathcal{X}_{k+1}||_\infty < \theta$ **then**
11: $\mathbf{p}_0 = \mathbf{p}_{k+1}$
12: return
13: **end if**
14: **end for**
15: Initial guess for next outer iteration $\mathbf{p}_0 = \mathbf{p}_P$
16: **end for**
17: **end procedure**

The algorithm terminates after all restarts have been completed, or when the maximum norm of the scaled gradient, $\|S^{-1}\nabla\mathcal{X}_k\|$, is smaller than the tolerance θ, $0 < \theta \ll 1$. In practice we often use $\theta = 10^{-12}$. The algorithm is given for a general preconditioning matrix S. When S is diagonal, $S^T S = S^2$.

The Polak–Ribière variant of the NLCG method is obtained by replacing the formula for β_k (line 7 in Algorithm 3) by

$$\beta_k = \frac{\nabla\mathcal{X}_{k+1}^T (S^T S)^{-1}(\nabla\mathcal{X}_{k+1} - \nabla\mathcal{X}_k)}{\nabla\mathcal{X}_k^T (S^T S)^{-1}\nabla\mathcal{X}_k}. \tag{14}$$

This method is restarted every Pth iteration and whenever β_k becomes zero or negative.

The preconditioned NLCG methods converges faster when the condition number of the scaled Hessian matrix is smaller. The scaled Hessian has elements $\hat{H}_{i,j} = \partial^2\mathcal{X}/\partial\hat{p}_i\partial\hat{p}_j$. Let H be the unscaled Hessian. In matrix notation, we have

$$\hat{H} = S^{-T} H S^{-1}, \quad H_{i,j} = \frac{\partial^2\mathcal{X}}{\partial p_i \partial p_j},$$

where S^{-T} denotes the transpose of the inverse of the scaling matrix S.

Both the BFGS and L-BFGS methods use a Newton-style update step

$$\mathbf{p}_{k+1} = \mathbf{p}_k - \tilde{H}_k^{-1} \nabla \mathcal{X}_k,$$

where $\tilde{H}_k$ is the approximation of the Hessian, which is updated along with $\mathbf{p}_k$. We refer to [25] for the precise update formula. No scaling is needed for these methods, since the Newton-style update step is scale invariant. However, both the BFGS and L-BFGS methods require an initial guess for the approximate Hessian, $\tilde{H}_0$. Here we use $\tilde{H}_0 = S^T S$, which makes $S^{-T} \tilde{H}_0 S^{-1} = I$, i.e., its scaled counterpart equals the identity matrix.

For many of the optimization algorithms considered here, convergence can not be guaranteed unless the Hessian exists and is a continuous function of the parameter vector, $\mathbf{p}$. In other words, the misfit functional $\mathcal{X}(\mathbf{p})$ must be twice continuously differentiable with respect to $\mathbf{p}$. It is straightforward to see that the displacement field depends linearly on the matrix elements of $\mathcal{M}$. Hence $\mathbf{u}$ and thereby $\mathcal{X}$ are infinitely differentiable with respect to the elements of $\mathcal{M}$. We assume that the time function depends on t_0 through a time shift, $g(t; t_0, \omega_0) = \tilde{g}(t - t_0; \omega_0)$. Because the source term $\mathbf{F}$ enters into the finite difference scheme with two time derivatives (see Algorithm 1), a requirement for the Hessian to be continuous is that $\tilde{g}(t; \omega_0)$ is four times differentiable with respect to t and twice differentiable with respect to ω_0. In Sect. 4, we derive a spatial discretization of the moment tensor source term that gives $\mathcal{X}$ the required regularity with respect to the source location, $\mathbf{x}_s$.

A crucial component of the NLCG methods is the line search algorithm, which seeks to minimize $\mathcal{X}(\mathbf{p}_k + \alpha \mathbf{q}_k)$ with respect to the step length α, see line 5 of Algorithm 3. Line search is also performed in the BFGS and L-BFGS methods, where the search direction is given by $\mathbf{q}_k = -(\tilde{H}_k)^{-1} \nabla \mathcal{X}_k$. For both types of methods, we use backtracking algorithm A6.3.1 from Dennis and Schnabel [7].

For the NLCG methods, the initial step size α_s is taken from the linear conjugate gradient algorithm, which assumes that $\mathcal{X}(\mathbf{p})$ is quadratic in $\mathbf{p}$. From that approximation follows

$$\alpha_s = -\frac{\nabla \mathcal{X}_k^T \mathbf{q}_k}{\mathbf{q}_k^T H_k \mathbf{q}_k}. \tag{15}$$

Here, the Hessian is evaluated at $\mathbf{p}_k$, i.e., $H_k = H(\mathbf{p}_k)$. In Sect. 3.2, we present an algorithm that evaluates $\mathbf{q}_k^T H_k \mathbf{q}_k$ by solving one additional wave equation.

3.1 The Gradient of the Misfit

Straightforward differentiation of (10) gives

$$\frac{\partial \mathcal{X}}{\partial p_j} = \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) \left\langle \mathbf{u}_{\mathbf{i}_r}^n - \mathbf{d}_r(t_n), \frac{\partial \mathbf{u}_{\mathbf{i}_r}^n}{\partial p_j} \right\rangle. \tag{16}$$

Note that the material properties ρ, μ, and λ do not depend on p_j. By differentiating the difference scheme for $\mathbf{u}$ with respect to p_j, we see that $\partial \mathbf{u} / \partial p_j$ can be calculated with the same finite difference scheme as is used for computing $\mathbf{u}$, if the source term $\mathbf{F}$ is replaced by $\partial \mathbf{F} / \partial p_j$. However, to compute the gradient of $\mathcal{X}$ with this technique, it would be necessary to solve the elastic wave equation with 11 different forcing functions, where each forcing corresponds to one component of $\partial \mathcal{X} / \partial p_j$.

A more efficient way of computing the gradient of the misfit is based on solving the adjoint wave equation. In this approach, we define the adjoint source in (12) as

$$\mathbf{G}_{\mathbf{i}}^{n} = \sum_{r=1}^{R} s(t_n) \left(\mathbf{u}_{\mathbf{i}_r}^{n} - \mathbf{d}_r(t_n) \right) \frac{\delta_{\mathbf{i},\mathbf{i}_r}}{h^3 a_{\mathbf{i}}}, \tag{17}$$

where $a_{\mathbf{i}}$ is the weight coefficient in the scalar product (8) and

$$\delta_{\mathbf{i},\mathbf{j}} = \begin{cases} 1, & \mathbf{i} = \mathbf{j}, \\ 0, & \text{otherwise.} \end{cases}$$

Inserting (17) into (16) shows that the gradient of the misfit can be written

$$\frac{\partial \mathcal{X}}{\partial p_j} = \sum_{n=0}^{M-1} \left(\mathbf{G}^n, \frac{\partial \mathbf{u}^n}{\partial p_j} \right)_h.$$

Because $\partial \mathbf{u}/\partial p_j$ satisfies the forward finite difference scheme with source term $\partial \mathbf{F}/\partial p_j$, we can apply Theorem 1 to obtain

$$\frac{\partial \mathcal{X}}{\partial p_j} = \sum_{n=0}^{M-1} \left(\boldsymbol{\kappa}^n, \frac{\partial \mathbf{F}^n}{\partial p_j} + \frac{\Delta_t^2}{12} \frac{\partial \mathbf{F}_{tt}^n}{\partial p_j} \right)_h + \frac{\Delta_t^2}{12} \sum_{n=0}^{M-1} \left(\boldsymbol{\zeta}^n, \frac{\partial \mathbf{F}^n}{\partial p_j} \right)_h. \tag{18}$$

Equation (18) allows us to calculate all components of the gradient from the adjoint wave field $\boldsymbol{\kappa}_{\mathbf{i}}^n$. The scalar products involving the gradients of $\mathbf{F}$ can be assembled during the time stepping of the adjoint scheme. Because the forcing function $\mathbf{F}$ is non-zero only at a few grid points near $\mathbf{x}_s$, the computational cost of evaluating these scalar products is insignificant compared to the cost of solving the adjoint wave equation.

3.2 Calculating the Hessian and $\mathbf{q}^T H \mathbf{q}$

The Hessian matrix plays an important role in gradient-based optimization. For example, the condition number of the Hessian governs the convergence rate of the conjugate gradient algorithm, and the Hessian can be used to construct a preconditioner.

To compute the Hessian, we differentiate (16) with respect to p_k to obtain

$$\begin{aligned} H_{k,j} &:= \frac{\partial}{\partial p_k} \left(\frac{\partial \mathcal{X}}{\partial p_j} \right) = \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) \frac{\partial}{\partial p_k} \left\langle \mathbf{u}_{\mathbf{i}_r}^{n} - \mathbf{d}_r(t_n), \frac{\partial \mathbf{u}_{\mathbf{i}_r}^{n}}{\partial p_j} \right\rangle \\ &= \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) \left\langle \frac{\partial \mathbf{u}_{\mathbf{i}_r}^{n}}{\partial p_k}, \frac{\partial \mathbf{u}_{\mathbf{i}_r}^{n}}{\partial p_j} \right\rangle + \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) \left\langle \mathbf{u}_{\mathbf{i}_r}^{n} - \mathbf{d}_r(t_n), \frac{\partial^2 \mathbf{u}_{\mathbf{i}_r}^{n}}{\partial p_k \partial p_j} \right\rangle. \end{aligned} \tag{19}$$

We decompose the Hessian into two parts, $H = H^{(1)} + H^{(2)}$, where

$$H_{k,j}^{(1)} := \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) \left\langle \frac{\partial \mathbf{u}_{\mathbf{i}_r}^{n}}{\partial p_k}, \frac{\partial \mathbf{u}_{\mathbf{i}_r}^{n}}{\partial p_j} \right\rangle, \tag{20}$$

$$H_{k,j}^{(2)} := \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) \left\langle \mathbf{u}_{\mathbf{i}_r}^{n} - \mathbf{d}_r(t_n), \frac{\partial^2 \mathbf{u}_{\mathbf{i}_r}^{n}}{\partial p_k \partial p_j} \right\rangle. \tag{21}$$

By noting the similarities between (16) and (21), we see that the matrix $H^{(2)}$ can also be computed using the adjoint wave field $\boldsymbol{\kappa}$. We arrive at the formula

$$H^{(2)}_{k,j} = \sum_{n=0}^{M-1} \left(\boldsymbol{\kappa}^n, \frac{\partial^2 \mathbf{F}^n}{\partial p_k \partial p_j} + \frac{\Delta_t^2}{12} \frac{\partial^2 \mathbf{F}^n_{tt}}{\partial p_k \partial p_j} \right)_h + \frac{\Delta_t^2}{12} \sum_{n=0}^{M-1} \left(\boldsymbol{\zeta}^n, \frac{\partial^2 \mathbf{F}^n}{\partial p_k \partial p_j} \right)_h . \tag{22}$$

Note that this formula is similar to (18), except that the first derivative of the forcing has been replaced by its second derivative. Hence, we can obtain $H^{(2)}$ by accumulating additional scalar products during the time stepping of the adjoint wave equation. Therefore, the computation of $H^{(2)}$ does not require any additional elastic wave equations to be solved. However, calculating $H^{(1)}$ requires the quantities $\partial \mathbf{u}^n_{\mathbf{i}} / \partial p_j$ to be known, which satisfy the elastic wave equation with the forcing term $\partial \mathbf{F} / \partial p_j$. Hence, an additional 11 elastic wave equations must be solved to assemble the matrix $H^{(1)}$.

The higher computational cost of calculating the Hessian makes it prohibitively expensive to evaluate in each iteration of Algorithm 3. However, as we will see below, it is highly advantageous to compute the Hessian at least once, and use it as a preconditioner throughout the iteration.

The step length calculation (15) for α_s requires the computation of the scalar quantity $\mathbf{q}^T H \mathbf{q}$, where $\mathbf{q}$ is a vector with 11 components. As before, we decompose the Hessian into $H = H^{(1)} + H^{(2)}$. The second term, $\mathbf{q}^T H^{(2)} \mathbf{q}$, is directly available after $H^{(2)}$ has been calculated, as described above. For the first term, we note that

$$\mathbf{q}^T H^{(1)} \mathbf{q} = \sum_{j=1,k=1}^{P} \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) \left\langle q_j \frac{\partial \mathbf{u}^n_{\mathbf{i}_r}}{\partial p_j}, \frac{\partial \mathbf{u}^n_{\mathbf{i}_r}}{\partial p_k} q_k \right\rangle .$$

Let $\tilde{\mathbf{u}}^n_{\mathbf{i}}$ denote the solution obtained by solving the discretized elastic wave equation with the forcing term $\sum_j q_j \frac{\partial \mathbf{F}(t_n)}{\partial p_j}$. It then holds that $\tilde{\mathbf{u}}^n_{\mathbf{i}} = \sum_j q_j \frac{\partial \mathbf{u}^n_{\mathbf{i}}}{\partial p_j}$, and hence,

$$\mathbf{q}^T H^{(1)} \mathbf{q} = \sum_{r=1}^{R} \sum_{n=0}^{M-1} s(t_n) \left\langle \tilde{\mathbf{u}}^n_{\mathbf{i}_r}, \tilde{\mathbf{u}}^n_{\mathbf{i}_r} \right\rangle, \tag{23}$$

can be assembled during the time stepping calculation of $\tilde{\mathbf{u}}^n$. The cost of calculating $\mathbf{q}^T H \mathbf{q}$ therefore amounts to solving one additional elastic wave equation. This is same cost as for computing the step length by an approximate difference quotient, as described in Kim et al. [15]. The advantage of using a step length based on (23) is that the errors in this difference approximation are avoided.

4 Discretizing the Singular Source Term

The gradient of the Dirac distribution in the seismic source term (2) is discretized based on the discretization of a one-dimensional Dirac distributions $\delta(x - x_s)$, and its derivative $\delta'(x - x_s)$. For all smooth, compactly supported functions of one variable $\varphi(x)$, we have

$$\int \varphi(x) \delta(x - x_s)\, dx = \varphi(x_s) \qquad \int \varphi(x) \frac{d\delta}{dx}(x - x_s)\, dx = -\frac{d\varphi}{dx}(x_s). \tag{24}$$

Our approach is based on the technique in Petersson and Sjögreen [26], which approximates the singular sources numerically by grid functions that satisfy (24) in a discrete scalar product for all polynomial functions up to order $q > 0$, leading to $q + 1$ moment conditions. The

required order is related to the order of accuracy of the approximation of the differential equation.

For a fourth order accurate scheme, the moment conditions for δ should be satisfied for the functions $\varphi(x) = x^k,\ k = 0, \ldots, 3$, and the moment conditions for δ' should be satisfied for $k = 0, \ldots, 4$. Details are given in Petersson and Sjögreen [26]. To make the technique easier to implement, we use discretizations that satisfy the moment conditions for $k = 0, \ldots, 4$, both for δ and δ'.

We describe the discretization of δ and its derivative in one space dimension. The multi-dimensional approximation can be obtained in a straightforward way by Cartesian products of the one-dimensional discretizations. Let the one-dimensional grid be $x_j = jh$, $j = 0, \ldots, N+1$, and define the scalar product by $(u, v)_{h1} = h \sum_{j=1}^{N} u_j v_j$. Furthermore, let the grid function $\tilde{b}_j = \tilde{b}(x_s, j_s)_j$ denote a preliminary approximation of $\delta(x - x_s)$, which is centered at grid point j_s. We make the straightforward choice $\tilde{b}_j = 0$ for $j < j_s - 2$ or $j > j_s + 2$, and determine the five coefficients in $\tilde{b}_j,\ j_s - 2 \le j \le j_s + 2$, by solving the system formed by the five moment conditions

$$\left(x^k, \tilde{b}\right)_{h1} = (x_s)^k, \quad k = 0, \ldots, 4. \tag{25}$$

The moment conditions on $\tilde{b}$ do not impose any specific relation between j_s and x_s. However, for accuracy reasons, we want to center the stencil near x_s. For example, we may choose j_s such that $x_{j_s} - h/2 \le x_s < x_{j_s} + h/2$. Within this interval, $\tilde{b}$ is infinitely differentiable with respect to x_s, because the elements $\tilde{b}_j$ are either zero, or depend on x_s through the right hand side of (25), which is a polynomial in x_s. However, if $x_s = x_{j_s} + h/2 + \varepsilon$, the stencil will be centered around grid point x_{j_s} for $\varepsilon < 0$, but around grid point x_{j_s+1} for $\varepsilon \ge 0$. Unfortunately, the elements of $\tilde{b}$ are not continuously differentiable with respect to x_s at $\epsilon = 0$, where the stencil switches center point.

The lack of continuity with respect to x_s can hamper the convergence of a gradient-based non-linear minimization algorithm. We will therefore replace $\tilde{b}(x_s, j_s)$ by a smoother source discretization, denoted by $b(x_s, j_s)$. The properties of $b(x_s, j_s)$ are given in the following theorem.

Theorem 2 *Let j_s be defined by $x_{j_s} \le x_s < x_{j_s+1}$, set $\nu = (x_s - x_{j_s})/h$, and let $\psi(\nu)$ be an m times continuously differentiable function having the properties*

$$\psi(0) = 0, \quad \psi(1) = 1, \quad \frac{d^l \psi}{d\nu^l}(0) = \frac{d^l \psi}{d\nu^l}(1) = 0, \quad l = 1, \ldots, m. \tag{26}$$

Furthermore, assume that $\tilde{b}(x_s, j_s)$ satisfies (25). Then, the grid function

$$b(x_s, j_s) = (1 - \psi(\nu))\, \tilde{b}(x_s, j_s) + \psi(\nu) \tilde{b}(x_s, j_s + 1), \tag{27}$$

is m times continuously differentiable with respect to x_s, and satisfies the moment conditions (25).

Proof Because $d/d\nu = h\, d/dx_s$, differentiability with respect to ν is equivalent with differentiability with respect to x_s. Also, $x_{j_s} \le x_s < x_{j_s+1}$ implies $0 \le \nu < 1$. As noted above, the elements of $\tilde{b}(x_s, j_s)$ and $\tilde{b}(x_s, j_s + 1)$ are polynomials in x_s, and are thus infinitely differentiable with respect to x_s. Because ψ is m times continuously differentiable, we conclude that b is m times continuously differentiable for $0 < \nu < 1$. At the stencil switching point $x_s = x_{j_s+1}$, the continuity conditions become

$$\frac{\partial^l b(x_s,\, j_s)}{\partial x_s^l}|_{x_s \to x_{j_s+1}} = \frac{\partial^l b(x_s,\, j_s+1)}{\partial x_s^l}|_{x_s = x_{j_s+1}}, \quad l = 0, \ldots, m. \tag{28}$$

Leibniz's product rule gives

$$\begin{aligned}\frac{\partial^l b(x_s,\, j_s)}{\partial x_s^l} &= (1-\psi(\nu))\frac{\partial^l \tilde{b}(x_s,\, j_s)}{\partial x_s^l} + \psi(\nu)\frac{\partial^l \tilde{b}(x_s,\, j_s+1)}{\partial x_s^l} \\ &\quad + \sum_{q=1}^{l} \binom{l}{q} \frac{1}{h^q} \frac{d^q \psi}{d\nu^q}(\nu) \left(\frac{\partial^{l-q} \tilde{b}(x_s,\, j_s+1)}{\partial x_s^{l-q}} - \frac{\partial^{l-q} \tilde{b}(x_s,\, j_s)}{\partial x_s^{l-q}} \right). \end{aligned} \tag{29}$$

The properties of ψ in (26) give, for $\nu = 0$,

$$\frac{\partial^l b(x_s,\, j_s)}{\partial x_s^l}|_{x_s = x_{j_s}} = \frac{\partial^l \tilde{b}(x_s,\, j_s)}{\partial x_s^l}|_{x_s = x_{j_s}}, \quad l = 0, 1, \ldots, m, \tag{30}$$

and for $\nu \to 1$,

$$\frac{\partial^l b(x_s,\, j_s)}{\partial x_s^l}|_{x_s \to x_{j_s+1}} = \frac{\partial^l \tilde{b}(x_s,\, j_s+1)}{\partial x_s^l}|_{x_s = x_{j_s+1}}, \quad l = 0, 1, \ldots, m. \tag{31}$$

Hence, by applying (30) to the source discretization centered at $j_s + 1$, we get

$$\frac{\partial^l b(x_s,\, j_s+1)}{\partial x_s^l}|_{x_s = x_{j_s+1}} = \frac{\partial^l \tilde{b}(x_s,\, j_s+1)}{\partial x_s^l}|_{x_s = x_{j_s+1}}, \quad l = 0, 1, \ldots, m. \tag{32}$$

The continuity conditions (28) now follow from (31) and (32).

The scalar product in the moment conditions (25) is computed by summation over the elements of the grid function b. This summation is clearly independent of j_s and x_s, so that we have

$$\begin{aligned}(x^k, b)_{h1} &= (1-\psi(\nu))\left(x^k, \tilde{b}(x_s,\, j_s)\right)_{h1} + \psi(\nu)\left(x^k, \tilde{b}(x_s,\, j_s+1)\right)_{h1} \\ &= (1-\psi(\nu))(x_s)^k + \psi(\nu)(x_s)^k = (x_s)^k,\end{aligned}$$

for $k = 0, \ldots, 4$. Therefore, $b(x_s,\, j_s)$ satisfies the moment condition (25). □

To construct a source discretization with two continuous derivatives, we apply (27) with the blending function

$$\psi(\nu) = \begin{cases} 0, & \nu < 0, \\ 10\nu^3 - 15\nu^4 + 6\nu^5, & 0 \le \nu < 1, \\ 1, & \nu \ge 1. \end{cases}$$

This function is monotonically increasing for $0 < \nu < 1$ and has two continuous derivatives at the break points $\nu = 0$ and $\nu = 1$. Since $\tilde{b}(x_s,\, j_s)$ is non-zero at five points, the grid function b has six non-zero elements. After some algebra, we find that the coefficients in the stencil (27) are given by

$$b(x_s, j_s)_{j_s-2} = \frac{1}{h}\left(\frac{1}{12}\nu - \frac{1}{24}\nu^2 - \frac{1}{12}\nu^3 - \frac{19}{24}\nu^4 + P(\nu)\right), \tag{33}$$

$$b(x_s, j_s)_{j_s-1} = \frac{1}{h}\left(-\frac{2}{3}\nu + \frac{2}{3}\nu^2 + \frac{1}{6}\nu^3 + 4\nu^4 - 5P(\nu)\right), \tag{34}$$

$$b(x_s, j_s)_{j_s} = \frac{1}{h}\left(1 - \frac{5}{4}\nu^2 - \frac{97}{12}\nu^4 + 10P(\nu)\right), \tag{35}$$

$$b(x_s, j_s)_{j_s+1} = \frac{1}{h}\left(\frac{2}{3}\nu + \frac{2}{3}\nu^2 - \frac{1}{6}\nu^3 + \frac{49}{6}\nu^4 - 10P(\nu)\right), \tag{36}$$

$$b(x_s, j_s)_{j_s+2} = \frac{1}{h}\left(-\frac{1}{12}\nu - \frac{1}{24}\nu^2 + \frac{1}{12}\nu^3 - \frac{33}{8}\nu^4 + 5P(\nu)\right), \tag{37}$$

$$b(x_s, j_s)_{j_s+3} = \frac{1}{h}\left(\frac{5}{6}\nu^4 - P(\nu)\right), \tag{38}$$

where

$$P(\nu) = \frac{5}{3}\nu^5 - \frac{7}{24}\nu^6 - \frac{17}{12}\nu^7 + \frac{9}{8}\nu^8 - \frac{1}{4}\nu^9,$$

and $b(x_s, j_s)_j = 0$ for all other j.

Let $e(x_s, j_s)_j$ denote the grid function approximating the derivative of the Dirac distribution, $\delta'(x - x_s)$. Following the same approach as above, we arrive at the six point stencil

$$e(x_s, j_s)_{j_s-2} = \frac{1}{h^2}\left(-\frac{1}{12} + \frac{1}{12}\nu + \frac{1}{4}\nu^2 + \frac{2}{3}\nu^3 + R(\nu)\right), \tag{39}$$

$$e(x_s, j_s)_{j_s-1} = \frac{1}{h^2}\left(\frac{2}{3} - \frac{4}{3}\nu - \frac{1}{2}\nu^2 - \frac{7}{2}\nu^3 - 5R(\nu)\right), \tag{40}$$

$$e(x_s, j_s)_{j_s} = \frac{1}{h^2}\left(\frac{5}{2}\nu + \frac{22}{3}\nu^3 + 10R(\nu)\right), \tag{41}$$

$$e(x_s, j_s)_{j_s+1} = \frac{1}{h^2}\left(-\frac{2}{3} - \frac{4}{3}\nu + \frac{1}{2}\nu^2 - \frac{23}{3}\nu^3 - 10R(\nu)\right), \tag{42}$$

$$e(x_s, j_s)_{j_s+2} = \frac{1}{h^2}\left(\frac{1}{12} + \frac{1}{12}\nu - \frac{1}{4}\nu^2 + 4\nu^3 + 5R(\nu)\right), \tag{43}$$

$$e(x_s, j_s)_{j_s+3} = \frac{1}{h^2}\left(-\frac{5}{6}\nu^3 - R(\nu)\right). \tag{44}$$

Here, the definition of ν and the relation between j_s and x_s are the same as for the grid function $b(x_s, j_s)$ above. The polynomial R is given by

$$R(\nu) = -\frac{25}{12}\nu^4 - \frac{3}{4}\nu^5 + \frac{59}{12}\nu^6 - 4\nu^7 + \nu^8,$$

and $e(x_s, j_s)_j = 0$ for $j < j_s - 2$ or $j > j_s + 3$.

It can be verified that the grid function e satisfies the moment conditions for a fourth order accurate discretization of $\delta'(x - x_s)$,

$$(1, e)_{h1} = 0 \qquad (x^k, e)_{h1} = -k(x_s)^{k-1} \quad k = 1, \ldots, 4,$$

and is twice continuously differentiable with respect to the source location x_s.

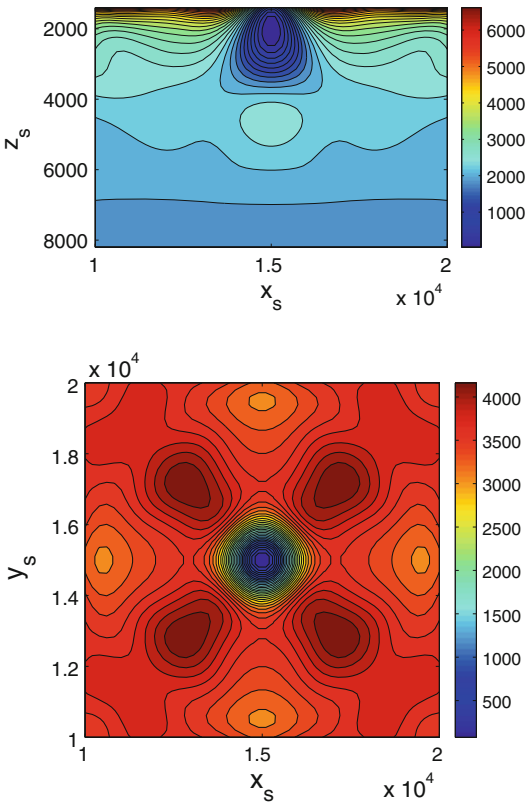


Fig. 1 Contour plots of $\mathcal{X}$ as function of the source location for the LOH.1 problem. *Top* $\mathcal{X}(x_s, z_s)$ for $y_s = 15{,}000$. *Bottom* $\mathcal{X}(x_s, y_s)$ for $z_s = 2{,}000$. All other parameters are fixed at the values corresponding to the global minimum

5 Estimating Initial Source Parameters

Figure 1 shows contour levels of $\mathcal{X}$ in two planes of the 11-dimensional parameter space, where the remaining nine parameters are held at their minimizing values. This example is taken from the layer over half space problem described in Sect. 6. The minimum is clearly visible at $x_s = y_s = 15{,}000$ and $z_s = 2{,}000$. Gradient-based minimization algorithms are derived under the assumption that the objective function is close to quadratic in parameter space. Figure 1 shows that this assumption only holds close to the minimum. Furthermore, the local minima in these cross-sections of parameter space indicate that $\mathcal{X}$ may have several local minima. To make the minimization algorithm converge to the global minimum, it follows that the initial parameter guess must be fairly accurate. We proceed by describing an approach for establishing initial parameter values for the source estimation problem.

5.1 Initial Estimate for the Source Location and Start Time

Our initial estimate for the source location is based on first arrival times, and variations of this technique are well-known in seismology [1]. For completeness, we give a brief description of the approach when the material is homogeneous. Assume that the first wave arrives at time t_r at receiver location (x_r, y_r, z_r). If the material has homogeneous properties with compressional wave speed c_p, the travel time from source location (x_s, y_s, z_s) to receiver r satisfies

$$\hat{T}_r(x_s, y_s, z_s) = \frac{1}{c_p}\sqrt{(x_r - x_s)^2 + (y_r - y_s)^2 + (z_r - z_s)^2}.$$

The source starting time, t_s, is related to the first arrival time, t_r, through $\hat{T}_r(x_s, y_s, z_s) + t_s = t_r$. Hence, we consider solving

$$\hat{T}_r(x_s, y_s, z_s) + t_s - t_r = 0, \quad r = 1, \ldots, R. \tag{45}$$

Because each receiver results in one equation for the four unknowns (x_s, y_s, z_s, t_s), we need at least four receivers. Usually there are more than four receivers, which makes (45) an overdetermined system. It can be solved in the least squares sense using the Gauss–Newton method. The technique is straightforward to generalize to a vertically layered material, but we omit the details to save space.

The above approach works well for synthetic data, but more work is needed to handle noisy measurements, in which case it can be difficult to precisely identify the arrival time.

5.2 Estimating the Source Frequency

It has turned out to be difficult to automatically estimate the source frequency parameter, ω_0. For this reason, we require an initial guess for ω_0 to be provided by the user. However, in practice this might not be a serious limitation, because in realistic applications the observed ground motions must be filtered in time to remove waves that can not be resolved on the computational grid. This is a preprocessing step that is performed before the optimization is started. The corner frequency of the filter is then related to the effective source frequency.

5.3 Initial Estimate for the Moment Tensor

Once initial estimates for the source location, frequency, and starting time have been established, we can use the linearity of the elastic wave equation to estimate the matrix $\mathcal{M}$ in the source term (2). Let $\mathbf{u}^{(xx)}$, $\mathbf{u}^{(xy)}$, $\mathbf{u}^{(xz)}$, $\mathbf{u}^{(yy)}$, $\mathbf{u}^{(yz)}$, and $\mathbf{u}^{(zz)}$ denote solutions of the elastic wave equation with the matrix $\mathcal{M}$ set to

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

respectively. The solution for a general $\mathcal{M}$ is then obtained as the linear combination

$$\begin{aligned} \mathbf{w}(m_{xx}, m_{xy}, m_{xz}, m_{yy}, m_{yz}, m_{zz}) := {} & m_{xx}\mathbf{u}^{(xx)} + m_{xy}\mathbf{u}^{(xy)} + m_{xz}\mathbf{u}^{(xz)} \\ & + m_{yy}\mathbf{u}^{(yy)} + m_{yz}\mathbf{u}^{(yz)} + m_{zz}\mathbf{u}^{(zz)}. \end{aligned}$$

The elements of $\mathcal{M}$ are determined by minimizing the wave form misfit

$$\mathcal{X} = \frac{1}{2}\sum_{r=1}^{R}\sum_{n=0}^{M-1} s(t_n)\left|\mathbf{w}_{\mathbf{i}_r}^n(m_{xx}, m_{xy}, m_{xz}, m_{yy}, m_{yz}, m_{zz}) - \mathbf{d}_{\mathbf{i}_r}^n\right|^2. \tag{46}$$

Because $\mathbf{w}$ is linear in m_{ij}, $\mathcal{X}$ is a quadratic function of m_{ij}. Its minimum can be computed by solving the 6×6 linear system $\partial\mathcal{X}/\partial m_{ij} = 0$.

6 Numerical Experiments

To verify our implementation and gain understanding of the performance of the proposed approach, it is convenient to conduct numerical experiments on synthetic data. We generate the synthetic data by solving the discretized elastic wave equation with given source parameters, and record the resulting motions at the receiver stations. In this way, the exact source parameters are known and we can easily evaluate the convergence properties of the minimization algorithm.

Our first test is a variation of the LOH.1 layer over half space problem, which originally was used to evaluate the accuracy of seismic wave propagation codes [6]. In this problem, a point moment tensor forcing with a Gaussian time function is applied at depth $z_s^* = 2{,}000\,\mathrm{m}$ below the free surface of a layered isotropic elastic material. The source time function is a Gaussian,

$$g(t; t_0, \omega_0) = \frac{\omega_0}{\sqrt{2\pi}} e^{-\omega_0^2 (t-t_0)^2/2},$$

which is parametrized by the frequency ω_0 and the center time t_0. In our version of the LOH.1 problem, the computational domain is a box of size $30{,}000 \times 30{,}000 \times 8{,}500$ m. Figure 2 shows the geometry of the problem and the material properties in SI-units. All computations use the grid spacing $h = 120\,\mathrm{m}$ and the elastic wave equation is integrated to time $T = 9\,\mathrm{s}$. The spatial grid has approximately 4.5 million points.

In the following numerical experiments, the 'measured' synthetic data is recorded at 25 receiver stations located on the free surface ($z = 0$), on a coarse 5×5 grid with spacing 3,000 m, see Fig. 2. The synthetic data is generated by solving the elastic wave equation with the source parameter vector $\mathbf{p}_*$, with components

$$\begin{aligned} x_s^* = y_s^* = 1.5 \cdot 10^4,\ z_s^* = 2 \cdot 10^3,\ m_{xy}^* = 10^{18}, \\ m_{xx}^* = m_{xz}^* = m_{yy}^* = m_{yz}^* = m_{zz}^* = 0,\ t_0^* = 1.45,\ \omega_0^* = 6.0. \end{aligned} \tag{47}$$

Compared to the standard LOH.1 problem [6], note that we have reduced the source frequency parameter ω_0 and increased t_0. These modifications are made to speed up the calculations by allowing the synthetic solution to be resolved on a coarser grid.

Initial approximations for the location and start time can be obtained from the first arrival ray tracing algorithm described in Sect. 5, applied to the material model shown in Fig. 2. With the user choice $\omega_0 = 6.3$, this procedure gives

$$x_s = 1.49808 \cdot 10^4,\ y_s = 1.50395 \cdot 10^4,\ z_s = 2.35273 \cdot 10^3,\ t_0 = 1.358.$$

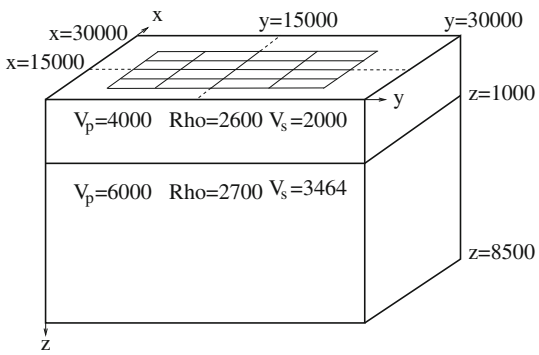


Fig. 2 The material properties in the layer over half space problem LOH.1. The receiver stations are placed on a 5 by 5 grid with spacing 3,000 m

We remark that the center time t_0 in the Gaussian time function follows from the source start time t_s as $t_0 = t_s + t_\delta$, where the half-duration is taken to be $t_\delta = 5/\omega_0$. After the location and center time have been determined, the moment tensor can be estimated by using the algorithm in Sect. 5.3, resulting in

$$m_{xx} = 3.754 \cdot 10^{14},\ m_{xy} = 9.622 \cdot 10^{17},\ m_{xz} = -8.937 \cdot 10^{15},$$
$$m_{yy} = 3.758 \cdot 10^{14},\ m_{yz} = 4.351 \cdot 10^{15},\ m_{zz} = -5.313 \cdot 10^{12}.$$

This initial parameter estimate is sufficiently accurate to make the minimization algorithms converge to the global minimum.

In our implementation of the source inversion algorithm, the user can either provide an initial approximation of the parameters, or let the solver estimate them automatically. The main drawback of the automated algorithm is that six elastic wave equations must be solved to calculate the moment tensor components. Hence, computational time can be saved if a sufficiently accurate approximation of the source parameters is already known, for example from a previous solution of a nearby problem.

Our practical experience is that the number of iterations required to reach convergence is not sensitive to the exact choice of initial guess. In many of the numerical experiments below, we use the following initial parameter values:

$$x_s = 1.6 \cdot 10^4,\ \ y_s = 1.4 \cdot 10^4,\ \ z_s = 2.2 \cdot 10^3,\ \ m_{xy} = 1.2 \cdot 10^{18},$$
$$m_{xx} = m_{xz} = m_{yy} = m_{yz} = m_{zz} = 0,\ \ t_0 = 1.54,\ \ \omega_0 = 6.3. \tag{48}$$

In Fig. 3 we plot the motion at two of the receiver stations, corresponding to the source parameters (47) and (48), respectively. Note how a relatively small change in source parameters leads to a significant change in ground motion at the receivers.

6.1 Scaling the Parameters

The sizes of the parameters in the source estimation problem span many orders of magnitude. In SI-units, $\mathbf{x}_s$ is of the order $\mathcal{O}(10^4)$, the moment tensor components m_{ij} are of the order $\mathcal{O}(10^{15}) - \mathcal{O}(10^{18})$. The parameters t_0 and ω_0 are both between $\mathcal{O}(1)$ and $\mathcal{O}(10)$. Because there is such a large difference in size between the smallest and largest parameter values, the original minimization problem is very poorly scaled and the condition number of the Hessian is very large.

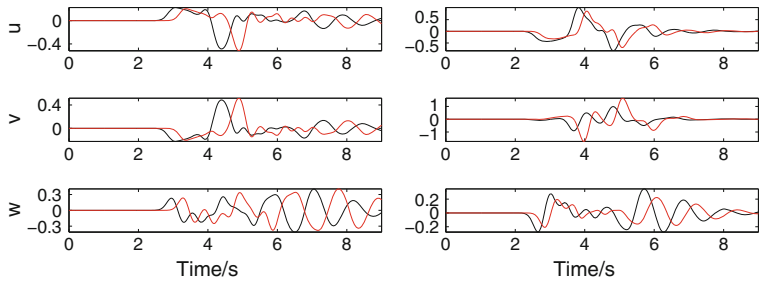


Fig. 3 The (u, v, w) components of the time-dependent ground motion at the receivers (x, y, z)=(9,000,21,000,0) (*left*) and (x, y, z)=(9,000,12,000,0) (*right*). Curves in *red* are generated with the initial source parameter values (48) and curves in *black* correspond to the solution of the minimization problem (47)

Scaling is important for both the quasi-Newton and the NLCG methods. For the quasi-Newton methods, the scaling is used to calculate the initial approximate Hessian, $\tilde{H}_0$. For the NLCG methods, the parameters should ideally be scaled such that the Hessian at the solution has condition number one. Let, $H_* := H(\mathbf{p}_*)$ denote the Hessian at the solution. The change of parameters $\hat{\mathbf{p}} = S\mathbf{p}$ gives the scaled Hessian $\hat{H}_* = (S^{-1})^T H_* S^{-1}$, i.e., the scaling corresponding to $\hat{H}_* = I$ satisfies

$$S^T S = H_*. \tag{49}$$

Hence, S could be computed by a Cholesky factorization of H_*. However, H_* is in general not computable because it can not be evaluated until $\mathbf{p}_*$ is known, i.e., after the minimization problem has been solved. Instead we can use a Cholesky factorization of the Hessian at the initial parameter guess. However, this Hessian is not guaranteed to be positive definite. As a result, its Cholesky decomposition may not always be well defined. Since the optimal scaling matrix can be difficult to determine and the implementation of the preconditioned NLCG algorithm is more straightforward when the scaling matrix S is diagonal, we will in the following only consider diagonal scalings. As we shall see, a significant reduction of the condition number of the Hessian can still be achieved. When S is diagonal, (49) can not be satisfied exactly. Instead we minimize the residual, $||H - S^2||_F$, which gives $S_{jj} = \sqrt{H_{jj}},\ j = 1, \ldots, P$. Hence, the scaling matrix should equal the square root of the diagonal of the Hessian. The Hessian at the minimum is positive definite, which implies that the diagonal elements of H_* are positive. Unfortunately, there is no guarantee that the Hessian at the initial guess is positive definite. If there are negative diagonal elements in the Hessian at the initial guess, we instead use the square root of the diagonal elements of the matrix H_1, see (20). It follows from the definition of H_1 that its diagonal elements always are non-negative.

The computation of the Hessian requires the elastic wave equation to be solved 11 times, see Sect. 3.1. However, this computation needs only to be done once because the same parameter scaling can be used throughout the minimization algorithm.

6.2 Condition Number of the Scaled Hessian

To evaluate the influence of different scalings, we calculate the condition number of the scaled Hessian for the LOH.1 source inversion problem, as computed by the Matlab function `cond`. The condition numbers of the scaled Hessian are given in the bottom row of Table 1. The Hessian is evaluated at the minimum, i.e., $H_* = H(\mathbf{p}_*)$, where $\mathbf{p}_*$ is given by (47). The diagonal variable transformation $\hat{\mathbf{p}} = S\mathbf{p}$ implies that the diagonal elements of S^{-1} can be interpreted as reference sizes for each of the parameters. However, note that only the ratio between the diagonal elements matter, because multiplying S by a constant factor does not change the condition number of the scaled Hessian.

The unscaled Hessian has condition number $\texttt{cond}(H_*) = 1.25 \cdot 10^{39}$. The second column of Table 1 shows the scaling obtained as the square root of the diagonal elements of the first part of the Hessian, H_1, evaluated at the initial parameter guess (48). Here we use H_1 because some diagonal elements of the full Hessian are negative. It is interesting to note that the scaling obtained from the square root of the diagonal of H_*, shown in column three, leads to a slightly larger condition number. The fourth column, labeled "Ref. size 1", shows the scaling based on estimated sizes of the parameters. Here we scaled x_s, y_s, and z_s by 10^4, t_0 by 1, ω_0 by 10, and all moment tensor components by 10^{18}. Table 1 shows that this scaling gives a significantly lower condition number compared to the unscaled case, but it is much larger compared to the scalings derived from either of the Hessians. After inspecting the Hessian-based scalings,

Table 1 Scaling factors and their influence on the condition number of the scaled Hessian. Here, H_1(48) corresponds to the first part of the Hessian evaluated at the parameter values (48). The condition number of the unscaled Hessian is 1.25×10^{39}

	H_1(48)	H_*	Ref. size 1	Ref. size 2	Ref. size 3
$1/s_{1,1}$ (x_s)	10.8	11.7	10^4	10^3	5×10^3
$1/s_{2,2}$ (y_s)	10.8	11.7	10^4	10^3	5×10^3
$1/s_{3,3}$ (z_s)	12.2	20.6	10^4	10^3	5×10^3
$1/s_{4,4}$ (m_{xx})	2.69×10^{16}	2.72×10^{16}	10^{18}	10^{18}	10^{18}
$1/s_{5,5}$ (m_{xy})	1.69×10^{16}	1.65×10^{16}	10^{18}	10^{18}	10^{18}
$1/s_{6,6}$ (m_{xz})	1.38×10^{16}	1.28×10^{16}	10^{18}	10^{18}	10^{18}
$1/s_{7,7}$ (m_{yy})	2.69×10^{16}	2.72×10^{16}	10^{18}	10^{18}	10^{18}
$1/s_{8,8}$ (m_{yz})	1.38×10^{16}	1.28×10^{16}	10^{18}	10^{18}	10^{18}
$1/s_{9,9}$ (m_{zz})	2.30×10^{16}	1.87×10^{16}	10^{18}	10^{18}	10^{18}
$1/s_{10,10}$ (t_0)	2.12×10^{-3}	2.56×10^{-3}	1	0.1	0.5
$1/s_{11,11}$ (ω_0)	5.65×10^{-2}	6.24×10^{-2}	10	1	5
$cond(S^{-1}H_*S^{-1})$	27.5	31.1	6.10×10^3	80.8	1.53×10^3

we modified the reference size scaling to be 10^3 for the location, 10^{18} for the moment tensor components, 0.1 for t_0, and 1 for ω_0. This scaling, labeled by "Ref. size 2" in Table 1, resulted in a significantly lower condition number. The last column of Table 1, labeled "Ref. size 3", shows a scaling that is in between "Ref. size 1" and "Ref. size 2". However, it leads to a condition number close to that of "Ref. size 1", indicating how sensitive the condition number is to the scaling matrix. Hence, even though it is possible to design a favorable scaling by orders of magnitude arguments, significant amounts of tuning can be needed before a good scaling matrix is found. Of the diagonal scalings considered here, we conclude that those based on the Hessian, or H_1, provide the most reliable way of reducing the condition number.

6.3 Convergence Rates for the Fletcher–Reeves Algorithm

We use the LOH.1 source inversion problem to evaluate the convergence of the Fletcher–Reeves NLCG algorithm for different parameter scalings. All iterations are started at the initial guess (48). Figure 4 shows convergence histories for the misfit and the relative max norm of the scaled gradient, i.e., the max norm of the gradient divided by the max norm of the initial gradient. Here, the relative norm is used to compensate for variations in sizes between the different scaling matrices. These computations are run for up to 10 restarts ($m = 10$ in Algorithm 3), with each restart cycle consisting of $P = 11$ inner iterations. The magenta curve shows the convergence history when S is taken as the square root of the diagonal elements of H_*. The results from using the square root of the diagonal elements of the first part of the Hessian, H_1, evaluated at (48), is shown by the red curve in Fig. 4. This approach results in an almost identical convergence history. The cyan and blue curves are obtained with scalings corresponding to the cases "Ref. size 1" and "Ref. size 2" in Table 1. Note that the "Ref. size 1" scaling performs as poorly as the unscaled NLCG method, shown in black. Neither of these approaches show any sign of convergence after 110 (inner) iterations. In contrast, the "Ref. size 2" scaling performs at least as well as either of the Hessian-based scalings. The intermediate scaling, corresponding to "Ref. size 3" and the green curve, leads to significantly better convergence than "Ref. size 1", despite the fact that the condition

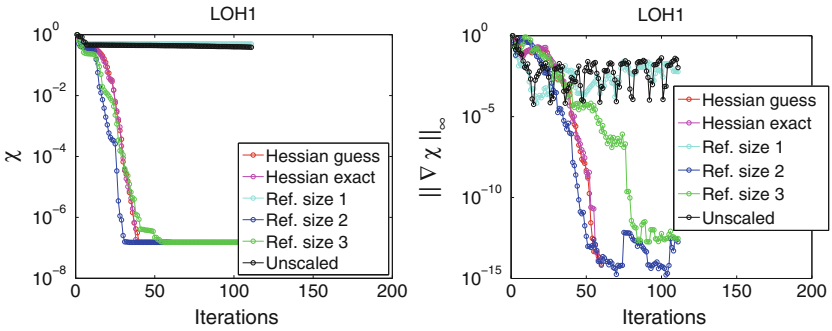


Fig. 4 Convergence of the misfit (*left*) and the relative maximum norm of the scaled gradient (*right*) for the different scalings given in Table 1

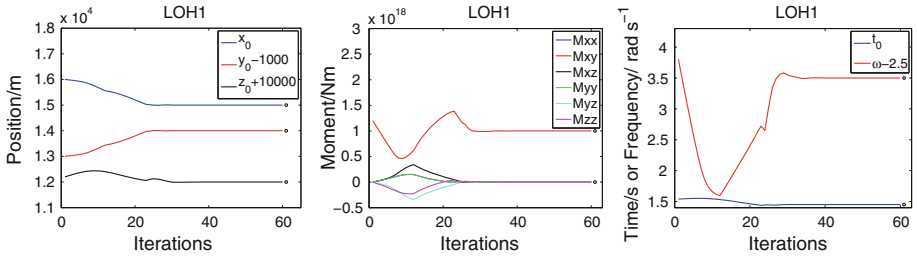


Fig. 5 Convergence of source parameters. Location (*left*), moment tensor components (*middle*), and time shift and frequency (*right*)

number of the scaled Hessian is only marginally smaller compared to "Ref. size 1". This illustrates that the condition number of the scaled Hessian can be a blunt indicator of the quality of a parameter scaling.

We conclude that the Hessian-based scalings always perform well, and the solution is obtained in 40–50 iterations with the Fletcher–Reeves algorithm. The reference size scalings can in principle be made equally effective, but the convergence rate is very sensitive to the exact values in the scaling matrix.

Figure 5 displays the evolution of the source parameters during the iterations, when the scaling is computed from the diagonal of H_1 at the initial guess. The left figure shows how the components of the source position (x_s, y_s, z_s) evolve during the iterations. Here, y_s is offset by 1,000 to distinguish it from x_s and z_s is offset by 10,000 to make it fit into the same plot. The circles to the right of the curves indicate the exact value of the parameter. Similarly, the middle subplot of Fig. 5 show the evolution of the six components of the matrix $\mathcal{M}$, and the right subplot shows the time shift and the frequency plotted with an offset. Note that all parameter values have converged in "picture norm" after about 33 iterations (3 outer iterations of the Fletcher–Reeves algorithm).

6.4 Comparing Optimization Methods

To compare the performance of the quasi-Newton and the NLCG algorithms, we again consider the LOH.1 source inversion problem, using (48) for the initial parameter values. For the NLCG method, we scale the parameters based on the diagonal of H_1, evaluated at the initial parameter guess. Both BFGS and L-BFGS require an initial guess for the approximate

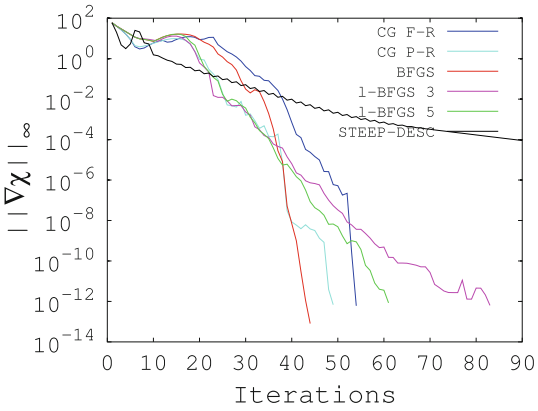


Fig. 6 Maximum norm of the scaled gradient during the solution of the LOH.1 source inversion problem using Fletcher–Reeves (*blue*), Polak–Ribière (*cyan*), BFGS (*red*), L-BFGS 3 (*magenta*), L-BFGS 5 (*green*), and steepest descent (*black*)

Hessian. For this purpose we use the diagonal of the matrix H_1, evaluated at the initial parameter values. This approach requires the same amount of computational work as to calculate the scaling for the NLCG methods. As a point of reference, we also evaluate the convergence of the steepest descent method, using the same diagonal scaling as for the NLCG methods.

The convergence histories for the NLCG and quasi-Newton methods are presented in Fig. 6, where we report the maximum norm of the scaled gradient during the iterations. Note the super-linear convergence rate close to the minimum for the BFGS and both NLCG methods. The L-BFGS method, using 3 or 5 vectors to represent the Hessian, show linear convergence near the minimum, and therefore require more iterations. Hence the BFGS and the NLCG methods require fewer iterations to determine the minimum very accurately. On the other hand, if the minimum only needs to be determined with moderate precision, e.g., by reducing the scaled gradient to $\mathcal{O}(10^{-4})$, most methods require approximately the same number of iterations. However, the Fletcher–Reeves method needs a few additional iterations to get to this level, and the steepest decent method is considerably slower.

The misfit and its gradient can be obtained by solving one elastic wave equation and one adjoint elastic wave equation. In the NLCG and the steepest descent methods, one additional elastic wave equation must be solved to calculate the initial step length (15). For all methods considered here, another elastic wave equation must be solved to test that the initial step length gives an adequate reduction of the misfit in the line search algorithm. One additional elastic wave equation must be solved per iteration of the backtracking algorithm. However, in most cases the initial step length is accepted and backtracking is only rarely invoked. If the initial step length is accepted, two elastic wave equations and one adjoint wave equation must be solved in each iteration of the BFGS and L-BFGS methods. The NLCG and steepest descent methods require three elastic wave equations and one adjoint wave equation to be solved per iteration.

Next we compare the computational performance for the different optimization methods. These calculations were performed with the parallel code SW4 [28] using 512 cores of Intel Xenon processors. In Table 2 we report the total CPU-time and total number of iterations, such that the max norm of the scaled gradient falls below 10^{-12}. The number of wave equations that must be solved in the NLCG or steepest decent methods, versus the quasi-Newton methods, predict a 4/3 ratio in execution time per iteration. Note that the calculations using L-BFGS are somewhat slower than expected. This is because backtracking was invoked more frequently for this method. We conclude that BFGS is the most computationally efficient

Table 2 Computational requirements of different optimization algorithms to reduce the maximum norm of the scaled gradient to 10^{-12}

Method	Number of iterations	Execution time (s)	Sec./iteration
Fletcher–Reeves	53	609	11.5
Polak–Ribière	48	560	11.7
BFGS	42	350	8.3
L-BFGS 3	82	840	10.2
L-BFGS 5	60	563	9.3
L-BFGS 11	46	466	10.1
Steepest descent	*	*	11.3

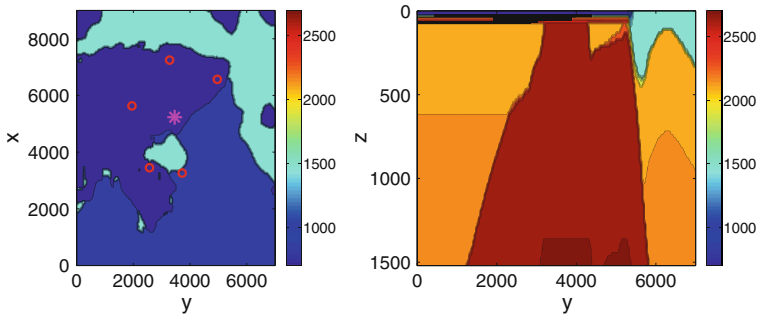


Fig. 7 S-velocity in the basic material model, on the free surface $z = 0$ (*left*) and on the vertical plane $x = 6{,}568$ (*right*). The *circles* and *star* in the *left* subplot indicate the receivers and source epicenter, respectively

method for solving this test problem. Of the NLCG methods, Polak–Ribière is about 10 % faster than Fletcher–Reeves. Steepest decent was the slowest of all methods, and did not meet the convergence criteria after 90 iterations.

6.5 An Example with a Realistic Material Model

We here demonstrate the source inversion technique in a domain with more complex material properties, modeling a region of the Nevada desert. The right-handed coordinate system is oriented to align the x axis with North, the y axis with East, and the z axis is directed vertically downwards, such that $z = 0$ corresponds to the free surface. The geometry of the problem is outlined in Fig. 7. The size of the computational domain in SI-units is $0 \leq x \leq 9{,}000$, $0 \leq y \leq 7{,}000$, and $0 \leq z \leq 1{,}520$, with grid size $h = 20$. We study the motion during the time interval $0 \leq t \leq 6$.

We generate synthetic 'measured' data by solving the elastic wave equation with a Gaussian source time function. The source parameters defining the minimum, $\mathbf{p}_*$, are given by

$$\begin{aligned} x_s^* = 5,232.016,\ y_s^* = 3,457.141,\ z_s^* = 45,\ \omega_0^* = 12,\ t_0^* = 0.5, \\ m_{xx}^* = m_{yy}^* = m_{zz}^* = 6 \cdot 10^{11}, \\ m_{xy}^* = m_{xz}^* = m_{yz}^* = 0. \end{aligned} \tag{50}$$

The synthetic displacement is recorded at five stations on the surface, marked by red circles in the left subplot of Fig. 7.

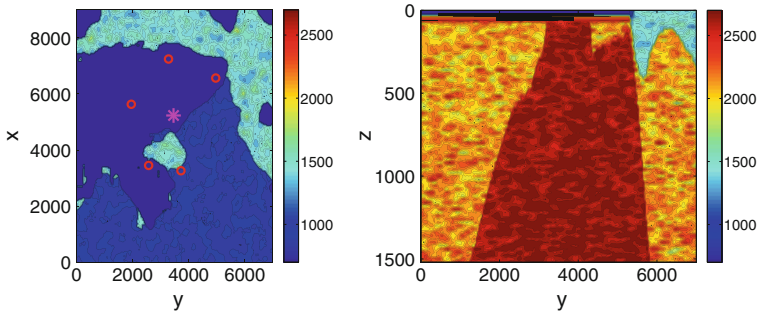


Fig. 8 S-velocity in the perturbed material model, on the free surface $z = 0$ (*left*) and on the vertical plane $x = 6{,}568$ (*right*)

Throughout this section, we use the BFGS method to solve the source inversion problems. As before, the initial approximate Hessian satisfies $\tilde{H}_0 = S^T S$, where the scaling S is calculated from the diagonal of the initial Hessian.

We start the source inversion from the perturbed source parameter values,

$$\begin{aligned} x_s = 5,232.016,\ y_s = 3,457.141,\ z_s = 50,\ \omega_0 = 11.5,\ t_0 = 0.55, \\ m_{xx} = 7.2 \cdot 10^{11},\ m_{yy} = 4.8 \cdot 10^{11},\ m_{zz} = 1.8 \cdot 10^{11}, \\ m_{xy} = -1.2 \cdot 10^{11},\ m_{xz} = 0,\ m_{yz} = 6 \cdot 10^{10}. \end{aligned} \tag{51}$$

The red curve in the left subplot of Fig. 9 shows the convergence history for the scaled gradient. The max norm of the scaled gradient falls below 10^{-12} after 44 iterations. At that point, the source parameters are within roundoff errors of the exact solution (50). This experiment indicates that our source inversion method also works in a more complex material model.

To test the robustness of our approach, we perturb the material wave speeds randomly such that the amplitude of the perturbation varies between 5 and 17 % of the unperturbed speed, with a spatial correlation length of 350 m in the horizontal directions and 60 m in the vertical direction. One realization of this procedure gives the material model shown in Fig. 8. We invert for the source parameters in the perturbed material based on the synthetic data from the unperturbed material model, using (51) as initial guess. The convergence history of the BFGS method is shown by the blue curve in the left subplot of Fig. 9. In this case, the max norm of the scaled gradient falls below 10^{-12} after 42 iterations. The inversion in the perturbed material model converges to a source with the following parameters:

$$\begin{aligned} x_s = 5,235.94,\ y_s = 3,453.54,\ z_s = 35.212,\ \omega_0 = 11.955,\ t_0 = 0.4999, \\ m_{xx} = 5.07098 \cdot 10^{11},\ m_{yy} = 5.135706 \cdot 10^{11},\ m_{zz} = 3.64143 \cdot 10^{11}, \\ m_{xy} = 1.699243 \cdot 10^{9},\ m_{xz} = -1.347843 \cdot 10^{9},\ m_{yz} = 4.764804 \cdot 10^{7}. \end{aligned} \tag{52}$$

We conclude that the source inversion method is well-conditioned in the sense that a moderate perturbation of the material leads to a moderate change in the source parameters. The solution appears to be the most sensitive to the source depth, z_s, and the m_{zz}-component of the moment tensor.

It is interesting to compare the wave forms corresponding to the source parameters (50) in the unperturbed material, and the optimized source parameters (52) in the perturbed material. In the right subplot of Fig. 9, we report the (u, v, w) components of the displacement at

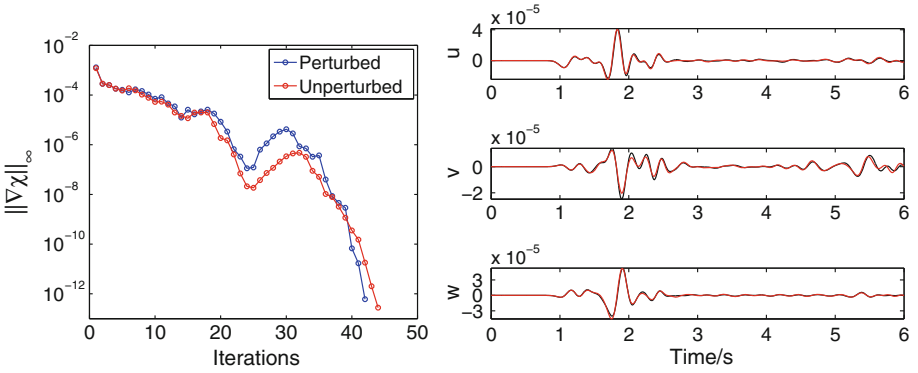


Fig. 9 *Left* convergence history for the max norm of the scaled gradient in the unperturbed material (*red*) and in the perturbed material (*blue*). *Right* (u, v, w) components of the ground motion in the unperturbed (*black*) and in the perturbed (*red*) material model, at receiver number 3, located at $x_3 = 3{,}260$, $y_3 = 3{,}720$, $z_3 = 0$

receiver number 3, located at $x_3 = 3{,}260$, $y_3 = 3{,}720$, $z_3 = 0$. In seismology, it is customary to decompose the horizontal motion in radial and transverse components relative to the source epicenter. For receiver # 3, the u and v components are close to being aligned with the radial and transverse directions, respectively. Note that the wave forms agree remarkably well, but some differences are visible in the transverse component. The behavior is similar at the other receiver locations.

7 Conclusions

We have presented an algorithm for estimating the seismic source parameters from recorded time dependent motions at a number of receiver stations. The solution of this inverse problem is obtained by minimizing the full waveform misfit using gradient-based optimization methods. The key features of the proposed technique are an adjoint discretization of the fourth order accurate method in Sjögreen and Petersson [31], a source discretization that leads to a misfit function that is twice continuously differentiable, and a parameter scaling that makes the minimization problem well conditioned. Numerical experiments on synthetic data indicate good convergence properties of the proposed algorithm.

We have compared the performance of the Fletcher–Reeves and Polak–Ribière conjugate gradient methods, and the BFGS and L-BFGS quasi-Newton algorithms. For a synthetic test case in a layered material model, all methods need about the same number of iterations to approximately locate the minimum. However, the conjugate gradient and BFGS methods converge faster near the solution and therefore need fewer iterations to determine the solution very accurately. Compared to the quasi-Newton methods, the conjugate gradient methods require an additional elastic wave equation to be solved per iteration to calculate the step length. As a result, we found the BFGS method to be the most efficient method for solving the source inversion problem, at least for the case considered here.

Several practical problems must be addressed before we can apply our algorithm to estimate source parameters for realistic seismic events. For example, the seismographic recordings must be preprocessed to compensate for instrument response characteristics, and the recorded signals must often be cleaned up to remove artifacts due to noise or other measurement errors. Additional filtering will be needed to remove frequencies that can not be resolved on the computational grid.

An interesting extension of the current approach is the inverse problem for estimating the material properties, i.e., the wave speeds and the density. Here, it would be desirable to find a suitable parametrization of the material that limits the dimensionality of parameter space. We expect that some degree of smoothness must be imposed on the material model, for example by using piecewise smooth basis functions to represent the material properties. Alternatively, a regularizing term could be added to the misfit functional.

Another interesting generalization of the inverse problem takes the uncertainty in the measured data into account and treats both the measurements and source parameters as probability distributions. See Tarantola [36] for a general discussion and Duputel et al. [10] for an application of these ideas to seismic source inversion.

Appendix 1: Proof of Theorem 1

We start by expanding the predictor into the corrector in Algorithm 1. Then rewrite the resulting expression as

$$\rho\frac{\mathbf{u}^{n+1}-2\mathbf{u}^{n}+\mathbf{u}^{n-1}}{\Delta_t^2}=\mathbf{L}_h(\mathbf{u}^n)+\mathbf{F}(t_n)+\frac{\Delta_t^2}{12}\big(\mathbf{L}_h(\mathbf{v}^n)+\mathbf{F}_{tt}(t_n)\big)+\mathbf{S}_G(\mathbf{u}^n-\mathbf{u}^{n-1}). \tag{53}$$

Next, take the scalar product between (53) and $\boldsymbol{\kappa}^n$, and sum over all time steps,

$$\sum_{n=0}^{M-1}\left(\boldsymbol{\kappa}^n,\rho\frac{\mathbf{u}^{n+1}-2\mathbf{u}^{n}+\mathbf{u}^{n-1}}{\Delta_t^2}\right)_h=\sum_{n=0}^{M-1}\left(\boldsymbol{\kappa}^n,\mathbf{L}_h(\mathbf{u}^n)\right)_h+\sum_{n=0}^{M-1}\left(\boldsymbol{\kappa}^n,\mathbf{F}(t_n)+\frac{\Delta_t^2}{12}\mathbf{F}_{tt}(t_n)\right)_h$$
$$+\frac{\Delta_t^2}{12}\sum_{n=0}^{M-1}\left(\boldsymbol{\kappa}^n,\mathbf{L}_h(\mathbf{v}^n)\right)_h+\sum_{n=0}^{M-1}\left(\boldsymbol{\kappa}^n,\mathbf{S}_G(\mathbf{u}^n-\mathbf{u}^{n-1})\right)_h. \tag{54}$$

The sum on the left hand side of (54) can be rewritten as

$$\sum_{n=0}^{M-1}\left(\boldsymbol{\kappa}^n,\rho\frac{\mathbf{u}^{n+1}-2\mathbf{u}^{n}+\mathbf{u}^{n-1}}{\Delta_t^2}\right)_h=\sum_{n=0}^{M-1}\left(\rho\frac{\boldsymbol{\kappa}^{n+1}-2\boldsymbol{\kappa}^{n}+\boldsymbol{\kappa}^{n-1}}{\Delta_t^2},\mathbf{u}^n\right)_h$$
$$+\frac{1}{\Delta_t^2}\left(\left(\rho\mathbf{u}^{-1},\boldsymbol{\kappa}^0\right)_h-\left(\rho\mathbf{u}^0,\boldsymbol{\kappa}^{-1}\right)_h+\left(\rho\mathbf{u}^M,\boldsymbol{\kappa}^{M-1}\right)_h-\left(\rho\mathbf{u}^{M-1},\boldsymbol{\kappa}^M\right)_h\right), \tag{55}$$

where the initial data $\mathbf{u}^0=\mathbf{u}^{-1}=\boldsymbol{\kappa}^M=\boldsymbol{\kappa}^{M-1}=\mathbf{0}$ make

$$\sum_{n=0}^{M-1}\left(\boldsymbol{\kappa}^n,\rho\frac{\mathbf{u}^{n+1}-2\mathbf{u}^{n}+\mathbf{u}^{n-1}}{\Delta_t^2}\right)_h=\sum_{n=0}^{M-1}\left(\rho\frac{\boldsymbol{\kappa}^{n+1}-2\boldsymbol{\kappa}^{n}+\boldsymbol{\kappa}^{n-1}}{\Delta_t^2},\mathbf{u}^n\right)_h.$$

The first sum on the right hand side of (54) is treated by the self-adjoint property,

$$\sum_{n=0}^{M-1}(\boldsymbol{\kappa}^n,\mathbf{L}_h(\mathbf{u}^n))_h=\sum_{n=0}^{M-1}(\mathbf{L}_h(\boldsymbol{\kappa}^n),\mathbf{u}^n)_h.$$

The second last sum of (54) can be rewritten

$$
\begin{aligned}
(\boldsymbol{\kappa}^n, \mathbf{L}_h(\mathbf{v}^n))_h &= (\mathbf{L}_h(\boldsymbol{\kappa}^n), \mathbf{v}^n)_h = \left(\mathbf{L}_h(\boldsymbol{\kappa}^n), \frac{1}{\rho}(\mathbf{L}_h(\mathbf{u}^n) + \mathbf{F}(t_n))\right)_h \\
&= \left(\mathbf{L}_h(\boldsymbol{\kappa}^n), \frac{1}{\rho}\mathbf{L}_h(\mathbf{u}^n)\right)_h + \left(\mathbf{L}_h(\boldsymbol{\kappa}^n), \frac{1}{\rho}\mathbf{F}(t_n)\right)_h \\
&= \left(\mathbf{L}_h\left(\frac{1}{\rho}\mathbf{L}_h(\boldsymbol{\kappa}^n)\right), \mathbf{u}^n\right)_h + \left(\frac{1}{\rho}\mathbf{L}_h(\boldsymbol{\kappa}^n), \mathbf{F}(t_n)\right)_h \\
&= (\mathbf{L}_h(\boldsymbol{\zeta}^n), \mathbf{u}^n)_h + (\boldsymbol{\zeta}^n, \mathbf{F}(t_n))_h. \qquad (56)
\end{aligned}
$$

The super-grid damping term can be written

$$
\begin{aligned}
\sum_{n=0}^{M-1} (\boldsymbol{\kappa}^n, \mathbf{S}_G(\mathbf{u}^n - \mathbf{u}^{n-1}))_h &= \sum_{n=0}^{M-1} (\boldsymbol{\kappa}^n, \mathbf{S}_G(\mathbf{u}^n))_h - \sum_{n=0}^{M-1} (\boldsymbol{\kappa}^{n+1}, \mathbf{S}_G(\mathbf{u}^n))_h \\
&\quad -(\boldsymbol{\kappa}^0, \mathbf{S}_G(\mathbf{u}^{-1}))_h + (\boldsymbol{\kappa}^M, \mathbf{S}_G(\mathbf{u}^{M-1}))_h. \qquad (57)
\end{aligned}
$$

The boundary terms are zero because of the initial data $\mathbf{u}^{-1} = \boldsymbol{\kappa}^M = 0$, and we use the symmetry (9) to obtain

$$
\sum_{n=0}^{M-1} (\boldsymbol{\kappa}^n, \mathbf{S}_G(\mathbf{u}^n - \mathbf{u}^{n-1}))_h = \sum_{n=0}^{M-1} (\boldsymbol{\kappa}^n - \boldsymbol{\kappa}^{n+1}, \mathbf{S}_G(\mathbf{u}^n))_h = -\sum_{n=0}^{M-1} (\mathbf{S}_G(\boldsymbol{\kappa}^{n+1} - \boldsymbol{\kappa}^n), \mathbf{u}^n)_h
$$

Collecting terms gives that (54) is equivalent to

$$
\begin{aligned}
\sum_{n=0}^{M-1} \left(\rho \frac{\boldsymbol{\kappa}^{n+1} - 2\boldsymbol{\kappa}^n + \boldsymbol{\kappa}^{n-1}}{\Delta_t^2}, \mathbf{u}^n\right)_h &= \sum_{n=0}^{M-1} (\mathbf{L}_h(\boldsymbol{\kappa}^n), \mathbf{u}^n)_h + \sum_{n=0}^{M-1} \left(\boldsymbol{\kappa}^n, \mathbf{F}(t_n) + \frac{\Delta_t^2}{12}\mathbf{F}_{tt}(t_n)\right)_h \\
+\frac{\Delta_t^2}{12} \sum_{n=0}^{M-1} (\mathbf{L}_h(\boldsymbol{\zeta}^n), \mathbf{u}^n)_h + \frac{\Delta_t^2}{12} \sum_{n=0}^{M-1} (\boldsymbol{\zeta}^n, \mathbf{F}(t_n))_h &- \sum_{n=0}^{M-1} (\mathbf{S}_G(\boldsymbol{\kappa}^{n+1} - \boldsymbol{\kappa}^n), \mathbf{u}^n)_h. \qquad (58)
\end{aligned}
$$

Expanding the predictor (11) into the corrector (12) gives

$$
\rho \frac{\boldsymbol{\kappa}^{n+1} - 2\boldsymbol{\kappa}^n + \boldsymbol{\kappa}^{n-1}}{\Delta_t^2} = \mathbf{L}_h(\boldsymbol{\kappa}^n) + \mathbf{G}(t_n) + \frac{\Delta_t^2}{12}\mathbf{L}_h(\boldsymbol{\zeta}^n) - \mathbf{S}_G(\boldsymbol{\kappa}^{n+1} - \boldsymbol{\kappa}^n). \qquad (59)
$$

Identity (13) of Theorem 1 is obtained by inserting (59) into the left hand side of (58).

References

1. Aki, K., Richards, P.G.: Quantitative Seismology, 2nd edn. University Science Books, Sausalito (2002)
2. Appelö, D., Colonius, T.: A high order super-grid-scale absorbing layer and its application to linear hyperbolic systems. J. Comput. Phys. **228**, 4200–4217 (2009)
3. Appelö, D., Petersson, N.A.: A stable finite difference method for the elastic wave equation on complex geometries with free surfaces. Commun. Comput. Phys. **5**, 84–107 (2008)
4. Berenger, J.P.: A perfectly matched layer for the absorption of electromagnetic waves. J. Comput. Phys. **114**, 185–200 (1994)
5. Cohen, G., Fauqueux, S.: Mixed spectral finite elements for the linear elasticity system in unbounded domains. SIAM J. Sci. Comput. **26**(3), 864–884 (2005)
6. Day, S.M., Bielak, J., Dreger, D., Larsen, S., Graves, R., Pitarka, A., Olsen, K.B.: Test of 3D elastodynamic codes: lifelines project task 1A01. Technical report, Pacific Earthquake Engineering Center (2001)
7. Dennis Jr, J.E., Schnabel, R.B.: Numerical Methods for Unconstrained Optimization and Nonlinear Equations. Prentice-Hall, Englewood Cliffs (1983)

8. Dumbser, M., Käser, M.: An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes—II. The three-dimensional isotropic case. Geophys. J. Int. **167**(1), 319–336 (2006)
9. Dumbser, M., Käser, M., de la Puente, J.: Arbitrary high-order finite volume schemes for seismic wave propagation on unstructured meshes in 2D and 3D. Geophys. J. Int. **171**(2), 665–694 (2007)
10. Duputel, Z., Rivera, L., Fukahata, Y., Kanamori, H.: Uncertainty estimations for seismic source inversions. Geophys. J. Int. **190**, 1243–1256 (2012)
11. Fichtner, A.: Full Waveform Modelling and Inversion. Advances in Geophysical and Environmental Mechanics and Mathematics. Springer, Berlin (2011)
12. Graves, R.W.: Simulating seismic-wave propagation in 3-D elastic media using staggered-grid finite differences. Bull. Seismo. Soc. Am. **86**(4), 1091–1106 (1996)
13. Jameson, A.: Aerodynamic design via control theory. J. Sci. Comput. **3**(3), 233–260 (1988)
14. Käser, M., Dumbser, M.: An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes—I. The two-dimensional isotropic case with external source terms. Geophys. J. Int. **166**(2), 855–877 (2006)
15. Kim, Y.H., Liu, Q., Tromp, J.: Adjoint centroid-moment tensor inversions. Geophys. J. Int. **186**(1), 264–278 (2011)
16. King, G.E.: Hydraulic fracturing 101: what every representative, environmentalist, regulator, reporter, investor, university researcher, neighbor and engineer should know about estimating frac risk and improving frac performance in unconventional gas and oil wells. Society of Petroleum Engineers, Feb (2012). SPE 152596
17. Kolda, T.G., Lewis, R.M., Torczon, V.: Optimization by direct search: new perspectives on some classical and modern methods. SIAM Rev. **45**, 385–482 (2003)
18. Komatitsch, D., Tromp, J.: Introduction to the spectral element method for three-dimensional seismic wave propagation. Geophys. J. Int. **139**, 806–822 (1999)
19. Lailly, P.: The seismic inverse problem as a sequence of before stack migrations. In: Bednar, J.B. (eds) Conference on Inverse scattering: Theory and Applications, pp. 206–220. Society of Industrial and Applied Mathematics, Philadelphia, PA (1983)
20. Levander, A.R.: Fourth-order finite-difference P-SV seismograms. Geophysics **53**, 1425–1436 (1988)
21. Lions, J.L.: Optimal Control of Systems Governed by Partial Differential Equations. Springer, Berlin (1971)
22. Liu, Q., Polet, J., Komatitsch, D., Tromp, J.: Spectral-element moment tensor inversions for earthquakes in southern California. Bull. Seismo. Soc. Am. **94**(5), 1748–1761 (2004)
23. Luenberger, D.G.: Introduction to Linear and Nonlinear Programming. Addison-Wesley, Reading (1973)
24. Nilsson, S., Petersson, N.A., Sjögreen, B., Kreiss, H.-O.: Stable difference approximations for the elastic wave equation in second order formulation. SIAM J. Numer. Anal. **45**, 1902–1936 (2007)
25. Nocecdal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer, Berlin (2006)
26. Petersson, N.A., Sjögreen, B.: Stable grid refinement and singular source discretization for seismic wave simulations. Commun. Comput. Phys. **8**(5), 1074–1110 (2010)
27. Petersson, N. A., Sjögreen, B.: Super-grid modeling of the elastic wave equation in semi-bounded domains. LLNL-JRNL 610212, Lawrence Livermore National Laboratory (2013). (submitted to Comm. Comput. Phys.)
28. Petersson, N.A., Sjogreen, B.: User's guide to SW4, version 1.0. LLNL-SM-xxyy, Lawrence Livermore National Laboratory (2013)
29. Pironneau, O.: On optimum design in fluid mechanics. J. Fluid Mech. **64**, 97–110 (1974)
30. Plessix, R.E.: A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. Geophys. J. Int. **167**, 495–503 (2006)
31. Sjögreen, B., Petersson, N.A.: A fourth order accurate finite difference scheme for the elastic wave equation in second order formulation. J. Sci. Comput. **52**, 17–48 (2012)
32. Skelton, E.A., Adams, S.D.M., Craster, R.V.: Guided elastic waves and perfectly matched layers. Wave Motion **44**, 573–592 (2007)
33. Song, F., Toksöz, M.N.: Full-waveform based complete moment tensor inversion and source parameter estimation from downhole seismic data for hydrofracture monitoring. Geophysics **76**(6), WC103–WC116 (2011)
34. Tape, C., Liu, Q., Tromp, J.: Finite-frequency tomography using adjoint methods - Methodology and examples using membrane surface waves. Geophys. J. Int. **168**, 1105–1129 (2007)
35. Tarantola, A.: Inversion of seismic reflection data in the acoustic approximation. Geophysics **49**, 1259–1266 (1984)
36. Tarantola, A.: Inverse Problem Theory and Methods for Model Parameter Estimation. SIAM, Philadelphia (2005)

37. Tromp, J., Komatitsch, D., Liu, Q.: Spectral-element and adjoint methods in seismology. Commun. Comput. Phys. **3**, 1–32 (2008)
38. Tromp, J., Tape, C., Liu, Q.: Seismic tomography, adjoint methods, time reversal and banana-doughnut kernels. Geophys. J. Int. **160**, 195–216 (2005)
39. Virieux, J.: P-SV wave propagation in heterogeneous media: velocity-stress finite-difference method. Geophysics **51**, 889–901 (1986)
40. Virieux, J., Operto, S.: An overview of full-waveform inversion in exploration geophysics. Geophysics **74**(6), WCC127–WCC152 (2009)

Super-grid modeling of the elastic wave equation in semi-bounded domains

N. Anders Petersson* and Björn Sjögreen*

January 18, 2013
Revised January 30, 2014

Abstract

We develop a super-grid modeling technique for solving the elastic wave equation in semi-bounded two- and three-dimensional spatial domains. In this method, waves are slowed down and dissipated in sponge layers near the far-field boundaries. Mathematically, this is equivalent to a coordinate mapping that transforms a very large physical domain to a significantly smaller computational domain, where the elastic wave equation is solved numerically on a regular grid. To damp out waves that become poorly resolved because of the coordinate mapping, a high order artificial dissipation operator is added in layers near the boundaries of the computational domain. We prove by energy estimates that the super-grid modeling leads to a stable numerical method with decreasing energy, which is valid for heterogeneous material properties and a free surface boundary condition on one side of the domain. Our spatial discretization is based on a fourth order accurate finite difference method, which satisfies the principle of summation by parts. We show that the discrete energy estimate holds also when a centered finite difference stencil is combined with homogeneous Dirichlet conditions at several ghost points outside of the far-field boundaries. Therefore, the coefficients in the finite difference stencils need only be boundary modified near the free surface. This allows for improved computational efficiency and significant simplifications of the implementation of the proposed method in multi-dimensional domains. Numerical experiments in three space dimensions show that the modeling error from truncating the domain can be made very small by choosing a sufficiently wide super-grid damping layer. The numerical accuracy is first evaluated against analytical solutions of Lamb's problem, where fourth order accuracy is observed with a sixth order artificial dissipation. We then use successive grid refinements to study the numerical accuracy in the more complicated motion due to a point moment tensor source in a regularized layered material.

1 Introduction

To numerically solve a time-dependent wave equation in an unbounded spatial domain, it is necessary to truncate the domain and impose a far-field closure at, or near, the boundaries of the truncated domain. Numerous different approaches have been suggested, see for

*Center for Applied Scientific Computing, L-422, LLNL, P.O. Box 808, Livermore, CA 94551, USA. This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This is contribution LLNL-JRNL-610212.

example [4, 6, 16]. The perfectly matched layer (PML) technique, originally proposed by Berenger [3] and later improved by many others, has been very successful for electromagnetic wave simulations. Unfortunately, the PML technique has stability problems when applied to the elastic wave equation, where free surface boundaries and material discontinuities can form wave guides in which the solution of the PML system becomes unstable [18]. The PML system is also known to exhibit stability problems for some anisotropic wave equations [2].

Similar to the PML technique, the super-grid method [1] modifies the original wave equation in layers near the boundary of the computational domain. The PML system is defined by Fourier transforming the original wave equation in time and applying a frequency-dependent complex-valued coordinate transformation in the layers. Additional dependent variables, governed by additional differential equations, must be introduced to define the PML system in the time domain. In comparison, the super-grid method is based on applying a real-valued coordinate stretching in the layers, where also artificial dissipation is added. The super-grid method does not rely on additional dependent variables, and is therefore more straight forward to implement. In the layers near the boundary, the PML method damps the waves; in contrast, the super-grid method both damps the waves and slows them down. The main advantage over the PML technique is that the solution of the wave equation with super-grid layers is energy stable, if there is a corresponding energy estimate for the underlying wave equation.

In this article, we generalize the super-grid approach [1] to the elastic wave equation in second order formulation. Motivated by applications from seismology and seismic exploration, we focus on half-plane or half-space domains, where a free surface boundary condition must be satisfied on only one side of the domain. The half-space problem subject to a free surface condition permits surface waves. These waves only propagate along the free surface and decay exponentially away from the surface. They are fundamentally different from the longitudinal and transverse waves that travel through the volume of the domain. Surface waves therefore constitute an additional type of wave that need to be absorbed by the far-field closure.

We are primarily interested in cases where the solution is of a transient nature, being driven by initial data with compact support, or by a forcing function that only is active (non-zero) for a limited time. Because of the artificial damping in the super-grid layers, the solution becomes very small on the outside of the layers. For this reason, it is natural to impose homogeneous Dirichlet conditions at the super-grid boundaries, which truncate the computational domain. In this paper, we develop a finite difference method where fourth order accurate summation by parts (SBP) operators [17] are combined with centered fourth order accurate finite difference formulas in the interior of the domain. The idea is to only use the SBP operators near the free surface boundary, and use centered finite difference formulas all the way up to the super-grid boundaries. This is made possible by enforcing homogeneous Dirichlet boundary conditions at several grid points outside the super-grid boundaries. This simplified boundary closure allows the implementation of the super-grid approach to be more efficient and greatly simplified in multi-dimensional domains, because the SBP operators are only needed near the free surface boundary.

SBP operators make it possible to prove stability for a difference approximation by mimicking the integration by parts estimate for the partial differential equation. In the interior of the domain, these operators are centered finite difference formulas. To satisfy the principle of SBP, the difference formulas must become biased and have coefficients

that are different for each grid point near a boundary. We call the discretization technique in [17] SBP-GP because it uses ghost points. These points are located just outside the boundary, and are used to enforce the boundary conditions strongly. There is also a related approach, called SBP-SAT [10, 9], which uses penalty terms to enforce the boundary conditions weakly. In principle, either of these SBP discretizations could be used to solve the elastic wave equation with super-grid layers.

The main theoretical result of this article is that stability of the numerical method can be proven also when the SBP operators are combined with centered operators, together with our simplified boundary closure near the super-grid boundaries. This leads to an overall spatial discretization that does not satisfy the principle of SBP, but nevertheless is energy stable. Note that the simplified boundary closure is intimately related to the super-grid method. It can only lead to an accurate approximation if the solution is very small near the super-grid boundary and is not appropriate for cases with inhomogeneous boundary data.

This paper continues the development of the super-grid method, with an emphasis on wave equations in second order formulation and multi-dimensional domains. While most of the development in [1] was done for hyperbolic systems in first order formulation in the continuous (PDE) setting, we focus on the elastic wave equation and establish stability results for the fully discrete approximation. We also present a new damping function for the super-grid layers, which gives the strongest damping in the outer parts of the supergrid layers. As a result, the suppression of outgoing waves is improved compared to the damping function used in [1]. A new tapering approach is introduced to scale the damping functions near edges and corners in multi-dimensional domains, such that the strength of the damping along the sides of the domain does not have to be reduced to meet the stability constraints of the explicit time integrator.

The remainder of the paper is organized in the follwing way. The super-grid method is first outlined in Section 1.1. In Section 2, we generalize our fourth order accurate time integration scheme [17] to the 1-D wave equation with grid stretching and artificial dissipation in the super-grid layers. We present our simplified boundary closure for the discretization that enables the centered difference stencils to be used all the way up to the super-grid boundaries. Because we solve the wave equation in second order formulation, the artificial dissipation contains a time derivative. We present an explicit time discretization and apply the energy method to prove stability of the fully discrete scheme. We outline a von Neumann analysis to show how the coefficient of the artificial dissipation must scale with the grid size to avoid a reduction of the explicit time step. It also exposes the stability limit of the amplitude of the damping function.

In Section 3 we generalize the results to the half-plane problem for the two-dimensional elastic wave equation subject to a free surface boundary condition along the physical boundary. The spatial discretization combines the SBP boundary modified stencils at the free surface boundary with our simplified boundary closure at the super-grid boundaries. The super-grid damping is introduced dimension by dimension and the energy method is used to show that the fully discrete scheme is stable. Our energy estimate shows that the strengths of the one-dimensional damping terms are accumulated near corners, where two super-grid layers meet. A two-dimensional von Neumann analysis illustrates that the explicit time step is limited by the sum of the strengths of the one-dimensional dissipation terms. To avoid having to either reduce the amplitude of the damping functions away from corners, or impose additional time step restrictions, we present a tapering approach

that reduces the strength of the one-dimensional damping functions near corners.

The reflection properties of the super-grid method are evaluated numerically by solving the three-dimensional elastic wave equation in Section 4. We first consider Lamb's problem, where the numerical solution is compared to an analytical solution. We also test the accuracy on a regularized layered material model, where the convergence is assessed by successive grid refinements. Conclusions are given in Section 5.

1.1 Outline of the supergrid method

Consider solving a time-dependent wave equation in an unbounded or semi-bounded (half-space) domain. Assume that we wish to calculate the numerical solution in a finite time interval $0 \le t \le t_{max}$, in the bounded spatial domain $\mathbf{x} \in \bar{\Omega} \in \mathrm{R}^d$ $(d = 1, 2, 3)$. We will call this the domain of interest. For $d = 3$ this could, for example, be a box shaped domain $\bar{\Omega} = \{x_1 \le x \le x_2,\ y_1 \le y \le y_2,\ z_1 \le z \le z_2\}$, where $\mathbf{x} = (x, y, z)^T$ are the Cartesian coordinates. Also assume that the initial conditions and forcing functions have compact support in $\bar{\Omega}$. While the numerical solution may be calculated in a computational domain that is larger than $\bar{\Omega}$, it must eventually be truncated to a finite extent. In general, the truncation of the computational domain leads to artificial reflections that can pollute the numerical solution in the domain of interest. However, due to the hyperbolic nature of wave equations, no artificial reflections can enter $\bar{\Omega}$ for $t \le t_{max}$, if the computational domain is sufficiently large. For example, if the computational domain is given by $x_1 - L \le x \le x_2 + L$, reflections from the outer boundary can only pollute the solution in the subdomain $x_1 \le x \le x_2$ for times $t > t_L = 2L/c_{max}$. Here, c_{max} is the largest phase velocity in the domain. Hence, by choosing $L \ge t_{max}\, c_{max}/2$, we can avoid all artifacts from the truncation of the computational domain, up to time $t = t_{max}$. Unfortunately, the size of the computational domain would grow with t_{max} and could easily become much larger than the original domain of interest. Hence, this simple approach is computationally intractable unless the grid size can be made significantly larger outside the domain of interest, without polluting the numerical solution with poorly resolved modes.

The first ingredient of the super-grid approach [1] is to introduce a smooth coordinate transformation,

$$x = X(\xi), \quad y = Y(\eta), \quad z = Z(\zeta),$$

that maps the computational domain onto a much larger extended domain. For example, in the x-direction, $x_1 - \ell \le \xi \le x_2 + \ell$ is mapped onto $x_1 - L \le x \le x_2 + L$, where $\ell \ll L$. The original wave equation is solved inside the domain of interest, i.e., the identity mapping $x = \xi$ is used for $x_1 \le \xi \le x_2$. The parts of the computational domain that are *outside* of the domain of interest are called the super-grid layers, i.e., $x_1 - \ell \le \xi < x_1$ and $x_2 < \xi \le x_2 + \ell$.

Spatial derivatives in the wave equation are transformed according to the chain rule,

$$\frac{\partial}{\partial x} = \phi^{(x)}(\xi)\frac{\partial}{\partial \xi}, \quad \frac{\partial}{\partial y} = \phi^{(y)}(\eta)\frac{\partial}{\partial \eta}, \quad \frac{\partial}{\partial z} = \phi^{(z)}(\zeta)\frac{\partial}{\partial \zeta}, \tag{1}$$

where

$$\phi^{(x)}(\xi) = \frac{1}{X'(\xi)}, \quad \phi^{(y)}(\eta) = \frac{1}{Y'(\eta)}, \quad \phi^{(z)}(\zeta) = \frac{1}{Z'(\zeta)}.$$

To make the coordinate transformation non-singular, we assume $\phi^{(q)} \ge \varepsilon_L > 0$, $q = x, y, z$.

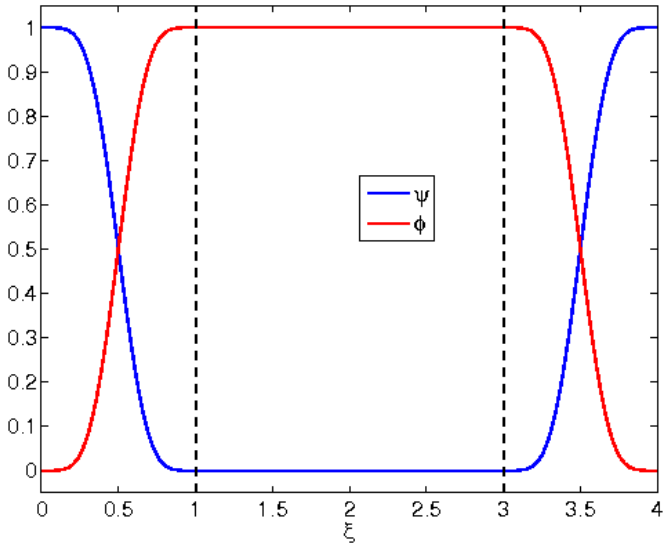


Figure 1: *The stretching function $\phi(\xi)$ (red) and the auxilliary function $\psi(\xi)$ (blue), which controls the strength of the damping. In this case, the width of each super-grid layer is $\ell = 1$. The dashed vertical lines indicate the boundaries of the domain of interest, here $1 \leq \xi \leq 3$.*

For a one-dimensional Cauchy problem, $\phi^{(x)}$ needs to be a smooth function that transitions monotonically from ε_L to 1 between $x_1 - \ell$ and x_1, and then back to ε_L between x_2 and $x_2 + \ell$, see Figure 1. For higher dimensional problems, the functions $\phi^{(y)}$ and $\phi^{(z)}$ are defined in a corresponding way.

In the mapped (computational) coordinates, the length scale of the solution in the ξ-direction is proportional to $\phi^{(x)}$. The solution is therefore compressed inside the layers, where $\phi^{(x)} < 1$. This corresponds to a slowing down of all traveling waves in the mapped coordinates. Note that in a two-dimensional domain, $\phi^{(x)} < 1$ corresponds to a slow down in the ξ-direction, while $\phi^{(y)} < 1$ gives a slow down in the η-direction. Hence, if the original wave equation has isotropic wave propagation properties, it becomes anisotropic in the mapped coordinates. The case of a half-plane problem in two space dimensions is illustrated in Figure 2.

The super-grid method discretizes the mapped (computational) domain on a grid with constant spacing. For this reason, the resolution in terms of grid points per wave length will be very poor in the layers. To avoid polluting the numerical solution by modes that can not be resolved on the grid, the second essential ingredient of the super-grid method is the addition of artificial damping. The dissipative term is only added in the layers and, for efficiency reasons, only explicit discretizations are considered. The idea is to damp out poorly resolved waves before they arrive at the outer edge of the layer, where the computational domain is truncated. As was emphasized in [1], it is important to use a damping term of sufficiently high order, such that it does not dominate the truncation error in the interior of the domain. The strength of the dissipation is controlled by the auxilliary function ψ (see Figure 1), which must be ramped up smoothly to avoid artificial reflections.

Figure 2: *A two-dimensional half-plane domain with a physical boundary along the top edge. The stretching functions satisfy* $\phi^{(x)} = \phi^{(y)} = 1$ *in the white region, where the original wave equation is solved. The wave speed is reduced in the surrounding layers by taking* $\phi^{(x)} < 1$ *(red) and* $\phi^{(y)} < 1$ *(blue). In the purple corner regions,* $\phi^{(x)} < 1$ *and* $\phi^{(y)} < 1$.

The amplitude of the damping must be large enough to damp out the solution before it is reflected back into the domain of interest. However, that amplitude is also restricted by the stability limit of the explicit time stepping scheme. Numerical experiments show that the super-grid method gives the best performance when the amplitude of the damping is close to the stability limit.

2 The scalar wave equation in one space dimension

Consider the Cauchy problem for the one-dimensional scalar wave equation,

$$\begin{aligned} \rho\frac{\partial^2 u}{\partial t^2} &= \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right) + f(x,t), \quad -\infty < x < \infty, \quad t \geq 0, \\ u(x,0) &= g_0(x), \quad u_t(x,0) = g_1(x), \quad -\infty < x < \infty. \end{aligned} \tag{2}$$

Here $\rho = \rho(x) > 0$ and $\mu = \mu(x) > 0$ are material coefficients that may vary in space, $g_0(x)$ and $g_1(x)$ are the initial data, and $f(x,t)$ is the external forcing function. The forcing and initial data are assumed to have compact support in the sub-domain $x \in \bar{\Omega}$ where $\bar{\Omega} = \{x_1 \leq x \leq x_2\}$. This is also assumed to be the domain of interest, i.e., where we want to find a numerical solution of (2).

We add a super-grid layer of width $\ell > 0$ on either side of $\bar{\Omega}$, and choose the coordinate system such that $x_1 - \ell = 0$ and $x_2 + \ell = x_{max}$. After introducing the coordinate mapping (1) (using the simplified notation $\phi = \phi^{(x)}$) and introducing a super-grid dissipation of order $2p$, we obtain the modified wave equation

$$\rho\frac{\partial^2 v}{\partial t^2} = \phi\frac{\partial}{\partial \xi}\left(\phi\mu\frac{\partial v}{\partial \xi}\right) - \varepsilon(-1)^p\phi\frac{\partial^p}{\partial \xi^p}\left(\sigma\rho\frac{\partial^p v_t}{\partial \xi^p}\right) + f(X(\xi),t), \tag{3}$$

for $0 \leq \xi \leq x_{max}$ and $t \geq 0$, where $\varepsilon > 0$. The solution of (3) is subject to the initial conditions

$$v(\xi,0) = g_0(X(\xi)), \quad v_t(\xi,0) = g_1(X(\xi)), \quad 0 \leq \xi \leq x_{max}. \tag{4}$$

The stretching function $\phi(x)$ and the damping function $\sigma(x)$ are constructed from the auxilliary function $\psi(x)$, which smoothly transitions from one to zero and then back to one,

$$\psi(x) = \begin{cases} 1, & x \le x_1 - \ell, \\ P((x_1 - x)/\ell), & x_1 - \ell < x < x_1, \\ 0, & x_1 \le x \le x_2, \\ P((x - x_2)/\ell), & x_2 < x < x_2 + \ell, \\ 1, & x \ge x_2 + \ell, \end{cases} \tag{5}$$

Here we use the polynomial function $P(\xi) = \xi^6(462 - 1980\xi + 3465\xi^2 - 3080\xi^3 + 1386\xi^4 - 252\xi^5)$, which satisfies $P(0) = 0$, $P(1) = 1$, and makes $\psi(\xi)$ five times continously differentiable. The one-dimensional stretching and damping functions are defined by

$$\phi(x) = 1 - (1 - \varepsilon_L)\psi(x), \quad \sigma(x) = \frac{\psi(x)}{\phi(x)}. \tag{6}$$

Note that the constant $\varepsilon_L > 0$ is not related to the damping coefficient ε in (3). Throughout the numerical experiments in this paper, we use $\varepsilon_L = 10^{-4}$. The functions ψ and ϕ are plotted in Figure 1.

The stretching and damping functions only modify the original wave equation inside the supergrid layers, because $\phi(x) = 1$ and $\sigma(x) = 0$, for $x_1 \le x \le x_2$. Also note the factor $\rho\sigma$ in the dissipation term in (3). Here, ρ is included to make this term balance the left hand side of the equation, i.e., to make ε independent of ρ.

We proceed by deriving an energy estimate. Note that the regular L_2 scalar product can be used to derive an energy estimate for the original wave equation (2). To estimate the solution of (3) it is therefore natural to weigh the scalar product by the stretching function (1). For real-valued functions $v(\xi)$ and $w(\xi)$, we define

$$(v, w)_\phi = \int_0^{x_{max}} \frac{v(\xi)\, w(\xi)}{\phi}\, d\xi, \quad \|v\|_\phi^2 = (v, v)_\phi,$$

where $\|v\|_\phi$ is a norm because $\phi \ge \varepsilon_L > 0$.

Assume that $f = 0$ and multiply the differential equation (3) by v_t/ϕ and integrate over $0 \le x \le x_{max}$. After integration by parts, we get

$$(v_t, \rho v_{tt})_\phi + (v_{t\xi}, \phi^2 \mu v_\xi)_\phi = -\varepsilon \left(\frac{\partial^p v_t}{\partial \xi^p}, \phi\sigma\rho \frac{\partial^p v_t}{\partial \xi^p} \right)_\phi + BT,$$

where the boundary term satisfies

$$BT = [v_t \phi \mu v_\xi]_0^{x_{max}} - \varepsilon(-1)^p \left[\left(v_t \frac{\partial^{p-1}}{\partial \xi^{p-1}} - \ldots + (-1)^{p-1} \frac{\partial^{p-1} v_t}{\partial \xi^{p-1}} \right) \left(\sigma\rho \frac{\partial^p v_t}{\partial \xi^p} \right) \right]_0^{x_{max}}. \tag{7}$$

The boundary term vanishes if we impose the following p boundary conditions at $\xi = 0$ and $\xi = x_{max}$,

$$\begin{aligned} v(0,t) &= 0, \quad & v(x_{max},t) &= 0, \\ v_\xi(0,t) &= 0, \quad & v_\xi(x_{max},t) &= 0, \\ &\vdots & &\vdots & t \ge 0. \\ \frac{\partial^{p-1} v}{\partial \xi^{p-1}}(0,t) &= 0, \quad & \frac{\partial^{p-1} v}{\partial \xi^{p-1}}(x_{max},t) &= 0. \end{aligned} \tag{8}$$

We define the energy by $E(t) = \frac{1}{2}(v_t, \rho v_t)_\phi + \frac{1}{2}(\phi v_\xi, \phi\mu v_\xi)_\phi$, which is a norm because $\rho > 0$, $\mu > 0$, and $\phi \geq \varepsilon_L > 0$. We arrive at

$$\frac{d}{dt}E(t) = -\varepsilon \left(\frac{\partial^p v_t}{\partial \xi^p}, \phi\sigma\rho \frac{\partial^p v_t}{\partial \xi^p} \right)_\phi \leq 0. \tag{9}$$

Hence, $E(t) \leq E(0)$ for $t > 0$. Assuming that the solution of the initial boundary value problem (3), (4), (8) exists[1], we conclude that is well-posed if $\varepsilon \geq 0$.

Because $E(t)$ is scaled by ρ, the strength of the damping in the super-grid layers is determined by the product $\varepsilon\phi\sigma$. This motivates our construction of $\sigma(x)$ in (6), which satisfies $\phi(x)\sigma(x) = \psi(x) \to 1$ for $x \to 0$ and $x \to x_{max}$. Our construction makes the damping the strongest in the outer parts of the super-grid layers. We remark that our construction is different from that used in [1], where the damping function satisfies $\sigma(x) \to 1$ for $x \to 0$ and $x \to x_{max}$. Because the strength of the damping is determined by the product $\sigma\phi$, the construction used in [1] gives a damping that is the strongest near the middle of the super-grid layers, and very weak in the outer parts of the super-grid layers, because $\phi\sigma \to \varepsilon_L \ll 1$ for $x \to 0$ and $x \to x_{max}$.

2.1 Discretizing the wave equation with super-grid layers

The theoretical properties of the discretization developed in this and the following sections builds to a large extent on the basic theory developed in [17]. Familiarity with that paper will expedite the understanding of the theory developed here.

We discretize the one-dimensional spatial domain on the uniform grid $\xi_j = (j-1)h$, $j = 1 - \tilde{p}, \ldots, 1, 2, 3, \ldots, N_x + \tilde{p}$, where $\tilde{p}$ is the number of ghost points (to be defined below). The constant grid spacing, $h > 0$, is chosen such that $\xi_{N_x} = x_{max}$. Time is discretized by $t_n = n\Delta_t$, where $n = 0, 1, 2, \ldots$ and $\Delta_t > 0$ is the constant time step. The value of the grid function u at the point (ξ_j, t_n) is denoted u_j^n. To simplify the notation, we occasionally drop the superscript or subscript on the grid function.

We discretize the spatial operator in (3) by the formula

$$\frac{\partial}{\partial \xi}\left(\phi\mu \frac{\partial u}{\partial \xi} \right)\bigg|_{\xi_j} = G(\phi\mu)u_j + \mathcal{O}(h^4),$$

where the fourth order centered operator is given by (here ϕ is absorbed into μ to simplify the notation),

$$\begin{aligned} G(\mu)u_j := \frac{1}{12h^2}\big(& \bar{\mu}_{j-1}(u_j - u_{j-2}) - 16\bar{\mu}_{j-1/2}(u_j - u_{j-1}) \\ & + 16\bar{\mu}_{j+1/2}(u_{j+1} - u_j) - \bar{\mu}_{j+1}(u_{j+2} - u_j)\big), \quad j = 1, 2, \ldots, N_x, \end{aligned} \tag{10}$$

and μ is averaged according to

$$\bar{\mu}_j = \frac{1}{2}\left(3\mu_{j-1} - 4\mu_j + 3\mu_{j+1}\right), \tag{11}$$

$$\bar{\mu}_{j+1/2} = \frac{1}{8}\left(\mu_{j-1} + 3\mu_j + 3\mu_{j+1} + \mu_{j+2}\right). \tag{12}$$

[1] Existence of solutions of the corresponding first order system with super-grid dissipation follows from Theorem 7.8.1 in [7]. Additional analysis would be required to establish existence of solutions of the wave equation considered here, but this is beyond the scope of the present article.

In [17], we developed a SBP boundary closure for (10). However, only the centered formula is needed here because no physical boundaries are present in the one-dimensional case.

A fourth order accurate time-integration scheme follows from the Taylor expansion

$$\frac{u_j^{n+1} - 2u_j^n + u_j^{n-1}}{\Delta_t^2} = u_{tt}|_j^n + \frac{\Delta_t^2}{12} u_{tttt}|_j^n + \mathcal{O}(\Delta_t^4). \tag{13}$$

We first consider the domain $x_1 \leq \xi \leq x_2$, where the super-grid dissipation term is zero because $\sigma = 0$. In that case, the semi-discrete approximation of (3) gives the formula for the second time derivative of u_j,

$$\rho_j u_{tt}|_j = \phi_j G(\phi\mu) u_j + f(\xi_j, t). \tag{14}$$

An expression for the fourth time derivative of u_j follows by differentiating (14) twice,

$$\rho_j u_{tttt}|_j = \phi_j G(\phi\mu) u_{tt}|_j + f_{tt}(\xi_j, t). \tag{15}$$

Substituting (14) and (15) into the Taylor series (13) gives the fourth order time stepping scheme in the interior of the domain.

When a fourth order ($p = 2$) artificial dissipation term is used in (3), it is discretized according to

$$(\sigma\rho\, u_{\xi\xi t})_{\xi\xi}|_j \approx D_+D_- \left(\sigma_j\rho_j\, D_+D_- \frac{u_j^n - u_j^{n-1}}{\Delta_t} \right) =: Q_4(\sigma\rho) \left(\frac{u_j^n - u_j^{n-1}}{\Delta_t} \right). \tag{16}$$

By replacing D_+D_- by $(D_+D_-)^{p/2}$, this formula generalizes to artificial dissipations of order $2p$, for $p = 0, 2, 4, \ldots$.

A sixth order ($p = 3$) artificial dissipation term is discretized according to

$$(\sigma\rho\, u_{\xi\xi\xi t})_{\xi\xi\xi}|_j \approx D_+D_-D_+ \left(\sigma_{j-1/2}\rho_{j-1/2}\, D_-D_+D_- \frac{u_j^n - u_j^{n-1}}{\Delta_t} \right) =: Q_6(\sigma\rho) \left(\frac{u_j^n - u_j^{n-1}}{\Delta_t} \right), \tag{17}$$

where the average is used for the coefficient, e.g., $\sigma_{j-1/2} = (\sigma_j + \sigma_{j-1})/2$. The above formula can be generalized to any odd $p \geq 1$ by replacing the difference operator $D_+D_-D_+$ by $(D_+D_-)^{(p-1)/2}D_+$, and $D_-D_+D_-$ by $D_-(D_+D_-)^{(p-1)/2}$.

Remark 1. *The discretizations of the dissipation terms are of low order accuracy. However, the coefficient ϵ will later be taken proportional to h^{2p-1}, thereby making these terms artificial in an approximation of the non-dissipative wave equation. With this choice of ϵ, and considering the scheme as an approximation of the non-dissipative wave equation, the dissipation term will restrict the overall accuracy to third order for $p = 2$, and to fifth order for $p = 3$.*

We arrive at the fully discrete approximation of (3),

$$\rho_j \frac{u_j^{n+1} - 2u_j^n + u_j^{n-1}}{\Delta_t^2} = \phi_j G(\phi\mu) u_j^n + f(\xi_j, t_n) + \frac{\Delta_t^2}{12} \left(\phi_j G(\phi\mu) \ddot{u}_j^n + f_{tt}(\xi_j, t_n) \right) - \varepsilon(-1)^p \phi_j Q_{2p}(\sigma\rho) \left(\frac{u_j^n - u_j^{n-1}}{\Delta_t} \right), \tag{18}$$

where

$$\ddot{u}_j^n = (\phi_j G(\phi\mu) u_j^n + f(\xi_j, t_n))/\rho_j. \tag{19}$$

The stencil for $G(\phi\mu)u_j$ in (10) is five points wide. If a fourth order dissipation is used, which also is five points wide, we must provide boundary conditions at two ghost points. Three ghost points are needed if a sixth order dissipation is used, because its stencil is seven points wide. In general, we need $\max(2, p)$ boundary conditions.

A natural discretization of the boundary conditions (8) is given by

$$B_{sg}(u^n) = (0, \ldots, 0)^T, \quad B_{sg}(\ddot{u}^n) = (0, \ldots, 0)^T, \quad n = 0, 1, 2, \ldots, \tag{20}$$

where the boundary operator $B_{sg}(u)$ picks out $\max(2, p)$ ghost point values outside each super-grid boundary,

$$B_{sg}(u) = (u_{1-\tilde{p}}, \ldots, u_0, u_{N_x+1}, \ldots, u_{N_x+\tilde{p}})^T, \quad \tilde{p} = \max(2, p). \tag{21}$$

Remark 2. *The implementation of the time-stepping scheme* (18), (19), (20) *can be simplified by writing it in predictor-corrector form [17]. This formulation also clairifies the application of the boundary conditions during one time step. Given u^n at all interior grid points, the ghost point values of u^n are first defined by enforcing $B_{sg}(u^n) = \mathbf{0}$. We can then evaluate* (19) *to compute $\ddot{u}^n$ at all interior grid points, after which its ghost point values are defined by enforcing $B_{sg}(\ddot{u}^n) = \mathbf{0}$. This defines the corrector term $G(\phi\mu)\ddot{u}^n$ and allows u^{n+1} to be updated at all interior grid points.*

2.2 Discrete energy estimate

We begin by defining the one-dimensional discrete L_2 scalar product and norm for real-valued grid functions v_j, w_j, by

$$(v, w)_{h1} = h \sum_{j=1}^{N_x} v_j w_j, \quad \|v\|_{h1}^2 = (v, v)_{h1}.$$

Essential properties of $G(\phi\mu)u_j$ are specified in the following lemma.

Lemma 1. *Let u and v be real-valued grid functions satisfying the boundary condition $B_{sg}(u) = \mathbf{0}$, $B_{sg}(v) = \mathbf{0}$, and let $\mu_j > 0$ and $\phi_j \geq \varepsilon_L > 0$ be the grid functions representing the material property and the stretching function, respectively. The spatial operator $G(\phi\mu)$, defined by* (10)*, satisfies*

$$(v, Gu)_{h1} = -K_0(v, u) \in \Re, \tag{22}$$

where the function $K_0(v, u)$ is bilinear, symmetric and positive definite, i.e., $K_0(v, u) = K_0(u, v)$ and $K_0(u, u) \geq \gamma \|u\|_{h1}^2$, $\gamma > 0$.

Proof. See Appendix A.1. □

The super-grid dissipation term satisfies a similar lemma.

Lemma 2. *Let the real-valued grid functions σ and ρ satisfy $\sigma_j \geq 0$ and $\rho_j > 0$. Furthermore, let u and v be real-valued grid functions that satisfy the boundary conditions $B_{sg}(u) = \mathbf{0}$ and $B_{sg}(v) = \mathbf{0}$. The super-grid dissipation operator $Q_{2p}(\sigma\rho)$, defined by* (16) *or* (17)*, satisfies*

$$(v, Q_{2p}u)_{h1} = (-1)^p C_0(v, u) \in \Re,$$

where the function $C_0(v,u)$ is bilinear, symmetric, and positive semi-definite, i.e., $C_0(v,u) = C_0(u,v)$ and $C_0(u,u) \geq 0$.

Proof. See Appendix A.2. □

To derive a discrete energy estimate for (18), it is convenient to work with grid functions that do not have ghost points. We therefore define grid functions $\bar{u}_j$ and $\bar{v}_j$ such that

$$\bar{u}_j = u_j, \quad \bar{v}_j = v_j, \quad 1 \leq j \leq N_x.$$

We also define square matrices K and C_{2p} such that,

$$K_0(u,v) = (\bar{u}, K\bar{v})_{h1}, \quad C_0(u,v) = (\bar{u}, C_{2p}\bar{v})_{h1}, \quad \text{if } B_{sg}(u) = \mathbf{0} \text{ and } B_{sg}(v) = \mathbf{0}.$$

Because the function $K_0(u,v)$ is symmetric and positive definite, $(\bar{u}, K\bar{v})_{h1} = (K\bar{u}, \bar{v})_{h1}$ and $(\bar{v}, K\bar{v})_{h1} > 0$ for all $\bar{v} \neq 0$, i.e. $K = K^T$ and $K > 0$. From (22) we have $(u, Gv)_{h1} = -(\bar{u}, K\bar{v})_{h1}$. By taking $u = 0$ except at one interior grid point where $u_j = 1$, we obtain a pointwise identity. The same procedure applies to the damping term Q_{2p}, and we conclude that

$$Gv = -K\bar{v} \quad \text{and} \quad Q_{2p}v = (-1)^p C_{2p}\bar{v}, \quad \text{if } B_{sg}(v) = \mathbf{0}. \tag{23}$$

We write the forcing in (18) in vector form as $F(t)$, with elements $F_j(t) = f(X(\xi_j), t)$, $j = 1, 2, \ldots, N_x$. Also introduce the diagonal matrices M and Φ with elements $M_{jj} = \rho_j$ and $\Phi_{jj} = \phi_j$, respectively. Because the acceleration satisfies the boundary conditions $B_{sg}(\ddot{u}) = \mathbf{0}$, there is a grid function without ghost points with elements $\ddot{\bar{u}}_j = \ddot{u}_j$, for $1 \leq j \leq N_x$, such that

$$\ddot{\bar{u}} = -M^{-1}\Phi K\bar{u} + M^{-1}F.$$

We summarize these results in the following Lemma.

Lemma 3. *The time-integration scheme* (18) *can be written in matrix form as*

$$\frac{1}{\Delta_t^2} M\left(\bar{u}^{n+1} - 2\bar{u}^n + \bar{u}^{n-1}\right) = -\Phi K\bar{u}^n + F(t^n) + \frac{\Delta_t^2}{12}\left(-\Phi K\ddot{\bar{u}}^n + F_{tt}(t_n)\right) - \frac{\varepsilon}{\Delta_t}\Phi C_{2p}\left(\bar{u}^n - \bar{u}^{n-1}\right), \quad n = 0, 1, 2, \ldots. \tag{24}$$

The matrices K and C_{2p}, defined by (23)*, are both symmetric; K is positive definite and C_{2p} is positive semi-definite. The matrices M and Φ are diagonal with positive elements.*

The solution of (24) is subject to the inital conditions

$$\bar{u}_j^0 = g_0(X(\xi_j)), \quad \bar{u}_j^{-1} = \tilde{g}_1(X(\xi_j)), \quad j = 1, 2, \ldots, N_x, \tag{25}$$

where $\tilde{g}_1$ depends on g_0 and g_1.

Our main result for the discretization of the one-dimensional wave equation with super-grid layers is formulated in the following theorem.

Theorem 1. *Let $\bar{u}^n$, $n = 0, 1, 2, \ldots$, be a solution of the time-integration scheme described in Lemma 3. Define the discrete energy by*

$$e^{n+1/2} := \frac{1}{\Delta_t^2}\left(\bar{u}^{n+1} - \bar{u}^n, \Phi^{-1}M(\bar{u}^{n+1} - \bar{u}^n)\right)_{h1} + \left(\bar{u}^{n+1}, K\bar{u}^n - \frac{\Delta_t^2}{12}KM^{-1}\Phi K\bar{u}^n\right)_{h1} - \frac{\varepsilon}{2\Delta_t}\left(\bar{u}^{n+1} - \bar{u}^n, C_{2p}\left(\bar{u}^{n+1} - \bar{u}^n\right)\right)_{h1}. \tag{26}$$

The discrete energy $e^{n+1/2}$ is a norm of the solution if the inequalities

$$2\left(\bar{w}, \Phi^{-1} M R_1 \bar{w}\right)_{h1} > \varepsilon \Delta_t \left(\bar{w}, C_{2p} \bar{w}\right)_{h1}, \tag{27}$$

$$\left(\bar{w}, \Phi^{-1} M R_2 \bar{w}\right)_{h1} > 0, \tag{28}$$

are satisfied for all vectors $\bar{w} \neq 0$. Here, $R_1 = P_1(\Delta_t^2 M^{-1}\Phi K)$ and $R_2 = P_2(\Delta_t^2 M^{-1}\Phi K)$, where P_1 and P_2 are the matrix polynomials,

$$P_1(A) := I - \frac{1}{4}A + \frac{1}{48}A^2, \quad P_2(A) := \frac{1}{4}A - \frac{1}{48}A^2. \tag{29}$$

If $\varepsilon \geq 0$ and $F(t) = 0$, the solution of (24) satisfies the energy estimate

$$e^{n+1/2} = e^{n-1/2} - \frac{\varepsilon}{2\Delta_t}\left(\bar{u}^{n+1} - \bar{u}^{n-1}, C_{2p}\left(\bar{u}^{n+1} - \bar{u}^{n-1}\right)\right)_{h1}, \tag{30}$$

which is non-increasing in n. The time-stepping scheme (24) is therefore stable if the time step satisfies the inequalities (27) and (28).

Proof. See Appendix A.3 □

Remark 3. *In [17] we used the Cayley-Hamilton theorem to analyze the stability of the non-dissipative version of the time-stepping scheme (24). When $\varepsilon = 0$ the inequalities (27) and (28) simplify to eigenvalue conditions, and one can prove that the scheme is stable under the time step restriction*

$$\Delta_t \leq \frac{2\sqrt{3}}{\max_j \sqrt{\kappa_j}}, \quad M^{-1}\Phi K \mathbf{e}_j = \kappa_j \mathbf{e}_j, \quad \varepsilon = 0.$$

Note that $M^{-1}\Phi K$ has the same spectrum as the symmetric positive definite matrix

$$M^{-1/2}\Phi^{1/2} K \Phi^{1/2} M^{-1/2}.$$

Hence all eigenvalues κ_j are real and positive.

Unfortunately, the energy estimate does not tell us how large the eigenvalues are, and therefore only says that the time-stepping scheme is stable if the time-step is sufficiently small.

2.3 Estimating the time step

In this section we outline how a von Neumann analysis can be used to estimate the stability limit of the time step. For this purpose, we assume constant stretching and dissipation coefficients, as well as constant material properties,

$$\sigma = \sigma_0 \geq 0, \quad \phi = \phi_0 \geq \varepsilon_L > 0, \quad \mu = \mu_0 > 0, \quad \rho = \rho_0 > 0.$$

To study the stability we assume that the external forcing is zero, $f(x,t) = 0$. When all coefficients are constant, the fully discrete scheme (18) simplifies to

$$u_j^{n+1} - 2u_j^n + u_j^{n-1} = \frac{\Delta_t^2 \phi_0^2 \mu_0}{\rho_0}\left(D_4 u_j^n + \frac{\Delta_t^2 \phi_0^2 \mu_0}{12\rho_0}(D_4)^2 u_j^n\right) - \varepsilon(-1)^p \sigma_0 \phi_0 \Delta_t \left(D_+ D_-\right)^p \left(u_j^n - u_j^{n-1}\right), \tag{31}$$

where $D_4 u_j = D_+ D_- u_j - \frac{h^2}{12}(D_+ D_-)^2 u_j$ is a fourth order accurate approximation of $u_{\xi\xi}$.

We start by estimating the stability limit for Δ_t without super-grid dissipation and set $\varepsilon = 0$. After a straighforward von Neumann analysis, we find that the necessary conditions for stability are satisfied if

$$0 \leq \frac{\Delta_t}{h}\sqrt{\frac{\phi_0^2 \mu_0}{\rho_0}} \leq \frac{3}{2}, \quad \varepsilon = 0. \tag{32}$$

Here, $\sqrt{\phi_0^2 \mu_0/\rho_0}$ is the local phase velocity. When ϕ, μ and ρ vary in space, (32) must be satisfied at every point in the domain. Note that the stretching function ϕ reduces the phase velocity in the super-grid layers.

In practice, effects from variable coefficients and boundary conditions can be taken into account by adjusting the coefficient 3/2 on the right hand side of (32). Let C_{CFL} denote this adjusted value, which can be determined by numerical experiments. We arrive at the standard CFL-type condition,

$$\frac{\Delta_t}{h} \leq \frac{C_{CFL}}{c_{max}}, \quad c_{max} = \max_{0 \leq \xi \leq x_{max}} \sqrt{\frac{\mu(\xi)}{\rho(\xi)}}, \quad \varepsilon = 0. \tag{33}$$

Next we study the stability limit inside the super-grid layer, where $\phi \ll 1$ and the local phase velocity is very small. For simplicity we focus on the dissipation term only and consider the stability of

$$u_j^{n+1} - 2u_j^n + u_j^{n-1} = -\varepsilon(-1)^p \sigma_0 \phi_0 \Delta_t \left(D_+ D_-\right)^p \left(u_j^n - u_j^{n-1}\right), \tag{34}$$

By introducing the un-divided difference operators $\Delta_\pm = hD_\pm$ and the scaled dissipation coefficient

$$\gamma_{2p} = \frac{\varepsilon \Delta_t}{h^{2p}}, \tag{35}$$

the simplified scheme (34) can be written as

$$u_j^{n+1} - 2u_j^n + u_j^{n-1} = -(-1)^p \gamma_{2p} \sigma_0 \phi_0 \left(\Delta_+ \Delta_-\right)^p \left(u_j^n - u_j^{n-1}\right). \tag{36}$$

This difference scheme is independent of the grid size and the time step. A von Neumann analysis can be used to show that the necessary conditions for stability of (36) are satisfied if

$$0 \leq \gamma_{2p} \sigma_0 \phi_0 \leq \frac{2}{4^p}. \tag{37}$$

In practice, we want to use the largest time step that makes the scheme stable without super-grid dissipation, i.e., satisfies (33) with equality. The maximum value of the damping coefficient γ_{2p} is then determined by numerical experiments on a coarse grid. Because Δ_t/h satisfies (33) with equality, the scaling (35) gives

$$\epsilon = \frac{\gamma_{2p} h^{2p}}{\Delta_t} = \frac{\gamma_{2p}}{C} h^{2p-1}, \quad C := \frac{\Delta_t}{h} = \frac{C_{CFL}}{c_{max}}, \quad C = \text{const}. \tag{38}$$

An important consequence is that a super-grid dissipation term of order $2p$ introduces an $\mathcal{O}(h^{2p-1})$ perturbation of the original wave equation. Also note that (37) indicates that the scaled dissipation coefficient needs to be reduced by a factor 4 when p is increased by one, e.g., when changing from fourth to sixth order super-grid dissipation.

3 The elastic wave equation

This section generalizes the super-grid technique to the half-plane problem for the elastic wave equation subject to a free surface boundary condition along the physical boundary. We describe a fourth order accurate discretization that combines SBP-GP operators at the free surface boundary, centered operators in the interior, and our simplified boundary closure at the super-grid boundaries. For clarity of presentation, the description and analysis are done in two space dimensions. It should be straightforward for the reader to generalize the results to the three-dimensional equations. See, for example, [13] for a second order accurate discretization of the three-dimensional equations.

Consider the time-dependent elastic wave equation in the two-dimensional half-plane $\mathbf{x} = (x, y) \in \Omega = \{-\infty < x < \infty,\ 0 \le y \le \infty\}$, governing the displacement with Cartesian components $\mathbf{u} = (u, v)^T$,

$$\begin{aligned} \rho u_{tt} &= ((2\mu+\lambda)u_x + \lambda v_y)_x + (\mu v_x + \mu u_y)_y + f^{(x)}, \\ \rho v_{tt} &= (\mu v_x + \mu u_y)_x + (\lambda u_x + (2\mu+\lambda)v_y)_y + f^{(y)}, \end{aligned} \quad \mathbf{x} \in \Omega,\ t \ge 0. \tag{39}$$

The heterogeneous isotropic material is characterized by the density $\rho(\mathbf{x}) > 0$, and the Lamé parameters $\lambda(\mathbf{x})$ and $\mu(\mathbf{x}) > 0$. In the following we assume $\lambda(\mathbf{x}) > 0$. Furthermore, $(f^{(x)}, f^{(y)})^T$ are the components of the external forcing functions. The displacement is subject to initial conditions

$$\mathbf{u} = \mathbf{g}_0, \quad \mathbf{u}_t = \mathbf{g}_1, \quad \mathbf{x} \in \Omega,\ t = 0, \tag{40}$$

where $\mathbf{g}_0$ and $\mathbf{g}_1$ are the initial data. The solution is subject to a normal stress condition on the physical boundary,

$$\begin{aligned} \mu(v_x + u_y) &= \tau^{(xy)}, \\ (2\mu+\lambda)v_y + \lambda u_x &= \tau^{(yy)}, \end{aligned} \quad -\infty < x < \infty,\ y = 0,\ t \ge 0, \tag{41}$$

where $\tau^{(yy)}$ and $\tau^{(xy)}$ are the boundary forcing functions. When $\tau^{(yy)} = \tau^{(xy)} = 0$ this boundary condition is often called a free surface, or traction free, condition.

Similar to the one-dimensional case, we want to calculate the solution of (39)-(41) in the sub-domain $\mathbf{x} \in \bar{\Omega} = \{x_1 \le x \le x_2,\ 0 \le y \le y_2\}$. The initial data, external forcing, and boundary forcing functions are assumed to have compact support in $\bar{\Omega}$. We add super-grid layers of thickness ℓ outside all sides of $\bar{\Omega}$, except $y = 0$. We choose the coordinate system such that $x_1 - \ell = 0$, $x_2 + \ell = x_{max}$, $y_2 + \ell = y_{max}$, and introduce the coordinate transformation (1). Because of the physical boundary condition (41) along $y = 0$, we use a stretching function in the η-direction that satisfies $\phi^{(y)} = 1$ for $0 \le \eta \le y_2$. Similar to the one-dimensional case, $\phi^{(x)} = 1$ for $x_1 \le \xi \le x_2$. See Figure 2 for a layout of this configuration.

After transforming the spatial derivatives in (39) and adding an artificial dissipation of order $2p$, we get the elastic wave equation with super-grid layers,

$$\begin{aligned} \rho u_{tt} = {} & \phi^{(x)} \frac{\partial}{\partial \xi} \left(\phi^{(x)} (2\mu+\lambda) u_\xi + \phi^{(y)} \lambda v_\eta \right) + \phi^{(y)} \frac{\partial}{\partial \eta} \left(\phi^{(x)} \mu v_\xi + \phi^{(y)} \mu u_\eta \right) - \\ & \epsilon (-1)^p \phi^{(x)} \frac{\partial^p}{\partial \xi^p} \left(\sigma^{(x)} \rho \frac{\partial^p u_t}{\partial \xi^p} \right) - \epsilon (-1)^p \phi^{(y)} \frac{\partial^p}{\partial \eta^p} \left(\sigma^{(y)} \rho \frac{\partial^p u_t}{\partial \eta^p} \right) + f^{(x)}, \end{aligned} \tag{42}$$

$$\rho v_{tt} = \phi^{(x)}\frac{\partial}{\partial\xi}\left(\phi^{(x)}\mu v_\xi + \phi^{(y)}\mu u_\eta\right) + \phi^{(y)}\frac{\partial}{\partial\eta}\left(\phi^{(x)}\lambda u_\xi + \phi^{(y)}(2\mu+\lambda)v_\eta\right) -$$
$$\epsilon(-1)^p\phi^{(x)}\frac{\partial^p}{\partial\xi^p}\left(\sigma^{(x)}\rho\frac{\partial^p v_t}{\partial\xi^p}\right) - \epsilon(-1)^p\phi^{(y)}\frac{\partial^p}{\partial\eta^p}\left(\sigma^{(y)}\rho\frac{\partial^p v_t}{\partial\eta^p}\right) + f^{(y)}. \quad (43)$$

Similar to the one-dimensional case, the coefficients in the damping terms are zero inside the domain of interest, i.e., $\sigma^{(x)}(\xi) = 0$ for $x_1 \le \xi \le x_2$ and $\sigma^{(y)}(\eta) = 0$ for $0 \le \eta \le y_2$. The damping in the ξ-direction is therefore only added in the layers $0 \le \xi \le \ell = x_1$ and $x_2 \le \xi \le x_2 + \ell = x_{max}$. In the η-direction, the damping is only added in the layer $y_2 \le \eta \le y_2 + \ell = y_{max}$. In particular, note that there is no damping in the η-direction near the physical boundary.

The normal stress boundary conditions (41) are also mapped to computational coordinates using (1). Because $\phi^{(y)} = 1$ for $y = \eta = 0$, we get

$$\begin{aligned} \mu\left(\phi^{(x)}v_\xi + u_\eta\right) &= \tau^{(xy)}, \\ (2\mu+\lambda)v_\eta + \lambda\phi^{(x)}u_\xi &= \tau^{(yy)}, \end{aligned} \qquad 0 \le \xi \le x_{max},\ \eta = 0,\ t \ge 0. \quad (44)$$

We proceed by deriving an energy estimate for the solution of (42), (43), (44). Because $\phi^{(x)} \ge \varepsilon_L > 0$ and $\phi^{(y)} \ge \varepsilon_L > 0$, we define a weighted scalar product and norm for real-valued functions u and v by

$$(u,v)_{2\phi} = \int_0^{y_{max}}\int_0^{x_{max}} \frac{u(\xi,\eta)\,v(\xi,\eta)}{\phi^{(x)}(\xi)\,\phi^{(y)}(\eta)}\,d\xi d\eta, \quad \|u\|_{2\phi}^2 = (u,u)_{2\phi}.$$

Consider the case without external and boundary forcing, i.e., $f^{(x)} = 0$, $f^{(y)} = 0$, $\tau^{(xy)} = 0$ and $\tau^{(yy)} = 0$. The energy estimate is derived by multiplying (42) by $u_t/(\phi^{(x)}\phi^{(y)})$, and (43) by $v_t/(\phi^{(x)}\phi^{(y)})$. We then add the results together and integrate over the computational domain. After integration by parts we obtain

$$\frac{d}{dt}E_2(t) = -\varepsilon\left(\phi^{(x)}\frac{\partial^p u_t}{\partial\xi^p}, \sigma^{(x)}\rho\frac{\partial^p u_t}{\partial\xi^p}\right)_{2\phi} - \varepsilon\left(\phi^{(x)}\frac{\partial^p v_t}{\partial\xi^p}, \sigma^{(x)}\rho\frac{\partial^p v_t}{\partial\xi^p}\right)_{2\phi} -$$
$$\varepsilon\left(\phi^{(y)}\frac{\partial^p u_t}{\partial\eta^p}, \sigma^{(y)}\rho\frac{\partial^p u_t}{\partial\eta^p}\right)_{2\phi} - \varepsilon\left(\phi^{(y)}\frac{\partial^p v_t}{\partial\eta^p}, \sigma^{(y)}\rho\frac{\partial^p v_t}{\partial\eta^p}\right)_{2\phi} + BT_2, \quad (45)$$

where BT_2 is the boundary term. In the stretched coordinates, the elastic energy satisfies

$$2E_2(t) = (u_t, \rho u_t)_{2\phi} + (v_t, \rho v_t)_{2\phi} + \left(\phi^{(x)}u_\xi + \phi^{(y)}v_\eta, \lambda\left(\phi^{(x)}u_\xi + \phi^{(y)}v_\eta\right)\right)_{2\phi} +$$
$$\left(\phi^{(x)}v_\xi + \phi^{(y)}u_\eta, \mu\left(\phi^{(x)}v_\xi + \phi^{(y)}u_\eta\right)\right)_{2\phi} +$$
$$2\left(\phi^{(x)}u_\xi, \mu\phi^{(x)}u_\xi\right)_{2\phi} + 2\left(\phi^{(y)}v_\eta, \mu\phi^{(y)}v_\eta\right)_{2\phi}. \quad (46)$$

Remark 4. *Similar to the one-dimensional case, the strength of the damping is determined by $\varepsilon\phi^{(x)}\sigma^{(x)}$ in the ξ-direction and by $\varepsilon\phi^{(y)}\sigma^{(y)}$ in the η-direction. The strength is accumulated near corners where two super-grid layers meet.*

The boundary term, BT_2, in (45) can be evaluated in the same way as was done for the one-dimensional case, in (7). Because $\tau^{(xy)} = \tau^{(yy)} = 0$ in boundary condition (44),

all boundary terms from $\eta = 0$ cancel. The remaining boundary terms in BT_2 become zero if we enforce the boundary conditions

$$\mathbf{u} = 0, \quad \mathbf{u}_\xi = 0, \cdots, \quad \frac{\partial^{p-1}\mathbf{u}}{\partial \xi^{p-1}} = 0, \quad \xi = \{0,\ x_{max}\},\ 0 \le \eta \le y_{max},\ t \ge 0, \tag{47}$$

and

$$\mathbf{u} = 0, \quad \mathbf{u}_\eta = 0, \cdots, \quad \frac{\partial^{p-1}\mathbf{u}}{\partial \eta^{p-1}} = 0, \quad 0 \le \xi \le x_{max},\ \eta = y_{max},\ t \ge 0. \tag{48}$$

The elastic energy $E_2(t)$ is a norm of $\mathbf{u}$ for all $\mathbf{u}$ that satisfy the homogeneous boundary conditions (44), (47), and (48), because $\rho > 0$, $\lambda > 0$, $\mu > 0$, $\phi^{(x)} \ge \varepsilon_L$, and $\phi^{(y)} \ge \varepsilon_L$, where $\varepsilon_L > 0$.

Remark 5. *Boundary conditions* (47) *and* (48) *remove the translational and rotational rigid body invariants from* $\mathbf{u}$. *These invariants would otherwise correspond to motions with zero elastic energy and make* $E_2(t)$ *a semi-norm, see e.g. [17] for details.*

We summarize the results of this section in the following lemma.

Lemma 4. *Let* $\mathbf{u} = (u, v)$ *be a solution of the elastic wave equation with super-grid dissipation* (42), (43), *subject to the boundary conditions* (44), (47), (48). *Let the order of the super-grid dissipation be* $2p$, $p \ge 0$. *Furthermore, assume that the external and boundary forcing functions are zero, i.e.* $f^{(x)} = f^{(y)} = 0$ *and* $\tau^{(xy)} = \tau^{(yy)} = 0$. *Furthermore, assume that the material parameters and the stretching functions satisfy* $\lambda > 0$, $\mu > 0$, $\rho > 0$, $\phi^{(x)} \ge \varepsilon_L$, *and* $\phi^{(y)} \ge \varepsilon_L$, *where* $\varepsilon_L > 0$. *Then, the elastic energy* $E_2(t)$, *defined by* (46), *is a norm of the solution and satisfies* (45) *with zero boundary term,* $BT_2 = 0$. *If the coefficient of the dissipation satisfies* $\varepsilon \ge 0$, *the right hand side of* (45) *is non-positive. Therefore,* $\mathbf{u}$ *satisfies the energy estimate* $E_2(t) \le E_2(0)$, *for* $t > 0$. *Assuming that the solution exists, we conclude that the problem is well-posed.*

3.1 Discretizing the elastic wave equation with super-grid layers

We discretize (42), (43) on the grid $\xi_i = (i-1)h$, $\eta_j = (j-1)h$, where i and j are integers. The domain sizes and the uniform grid spacing $h > 0$ are defined such that $x_{N_x} = x_{max}$ and $y_{N_y} = y_{max}$. Time is discretized on a grid with constant time step, $\Delta_t > 0$. We denote the approximation of the displacement at grid point (x_i, y_j) and time level $t_n = n\Delta_t$ by $\mathbf{u}^n_{i,j} = (u^n_{i,j}, v^n_{i,j})^T$.

The first two terms on the right hand sides of (42) and (43) are discretized according to

$$L^{(u)}_h \mathbf{u} = \phi^{(x)} G^{(x)} \left(\phi^{(x)} (2\mu + \lambda) \right) u + \phi^{(x)} D^{(x)} (\phi^{(y)} \lambda D^{(y)} v) + \\ \phi^{(y)} D^{(y)} (\phi^{(x)} \mu D^{(x)} v) + \phi^{(y)} G^{(y)} \left(\phi^{(y)} \mu \right) u, \tag{49}$$

and

$$L^{(v)}_h \mathbf{u} = \phi^{(x)} G^{(x)} \left(\phi^{(x)} \mu \right) v + \phi^{(x)} D^{(x)} (\phi^{(y)} \mu D^{(y)} u) + \\ \phi^{(y)} D^{(y)} (\phi^{(x)} \lambda D^{(x)} u) + \phi^{(y)} G^{(y)} \left(\phi^{(y)} (2\mu + \lambda) \right) v, \tag{50}$$

respectively. Here, the grid indices on the grid functions are suppressed to simplify the notation. On vector notation, the discretization is denoted

$$\mathbf{L}_h \mathbf{u}_{i,j}^n = \begin{pmatrix} L_h^{(u)} \mathbf{u}_{i,j}^n \\ L_h^{(v)} \mathbf{u}_{i,j}^n \end{pmatrix}.$$

The finite difference operators $G^{(x)}$ and $D^{(x)}$ in the above formulas act along the first index (ξ-direction). The fourth order accurate operator $G^{(x)}(\mu) w_{i,j}$ approximates $(\mu w_\xi)_\xi(\xi_i, \eta_j)$. Besides operating on a two-dimensional grid function, it is the same as the one-dimensional operator G in (10). The difference operator $D^{(x)} w_{i,j}$ is a fourth order accurate centered approximation of $w_\xi(\xi_i, \eta_j)$. It can be written

$$D^{(x)} w_{i,j} := D_{0x} w_{i,j} - \frac{h^2}{6} D_{0x} D_{+x} D_{-x} w_{i,j} = \\ \frac{1}{12h} \left(-w_{i+2,j} + 8 w_{i+1,j} - 8 w_{i-1,j} + w_{i-2,j} \right), \quad D_{0x} = \frac{1}{2}(D_{+x} + D_{-x}). \quad (51)$$

Note that the difference operators $G^{(x)}$ and $D^{(x)}$ are *not* boundary modified and do *not* satisfy standard SBP properties. As in the one-dimensional case, two ghost points are needed outside the super-grid boundaries $\xi = 0$ and $\xi = x_{max}$.

The fourth order accurate finite difference operators $G^{(y)}(\phi^{(y)}\mu)u$ and $D^{(y)}u$ approximate $(\phi^{(y)}\mu u_\eta)_\eta$ and u_η, respectively. These are one-dimensional operators acting along the second index (η-direction), but with SBP-GP boundary modifications at the $\eta = 0$ boundary, as described in [17]. For this reason, one ghost point is needed outside the physical boundary $\eta = 0$, and two ghost points are needed outside the super-grid boundary $\eta = y_{max}$, where there is no boundary modification.

The artificial dissipation operators in (42) and (43) are discretized in the same way as in the one-dimensional case. The dissipation of order $2p$ is denoted by $Q_{2p}^{(x)}$ in the ξ-direction and $Q_{2p}^{(y)}$ in the η-direction. On vector form, the two-dimensional dissipation becomes

$$\mathbf{Q}_{2p}\mathbf{u} = \begin{pmatrix} \phi^{(x)} Q_{2p}^{(x)}(\sigma^{(x)}\rho)u + \phi^{(y)} Q_{2p}^{(y)}(\sigma^{(y)}\rho)u \\ \phi^{(x)} Q_{2p}^{(x)}(\sigma^{(x)}\rho)v + \phi^{(y)} Q_{2p}^{(y)}(\sigma^{(y)}\rho)v \end{pmatrix}. \quad (52)$$

The dissipation requires p ghost points outside each super-grid boundary. Note that the dissipation in the y-direction does not need any ghostpoints outside $\eta = 0$, because the dissipation coefficient is zero near this boundary, i.e. $\sigma^{(y)} = 0$.

The normal stress boundary conditions (44) are discretized by the fourth order accurate formulas

$$\mu_{i,1} \left(B^{(y)} u_{i,1} + \phi_i^{(x)} D^{(x)} v_{i,1} \right) = \tau_i^{(xy)}, \quad (53)$$

$$(2\mu + \lambda)_{i,1} B^{(y)} v_{i,1} + \lambda_{i,1} \phi_i^{(x)} D^{(x)} u_{i,1} = \tau_i^{(yy)}, \quad (54)$$

for $1 \le i \le N_x$. The boundary operator $B^{(y)} v_{i,1}$ is derived in [17]. It is a fourth order accurate approximation of $v_y(x_i, y_1)$ of the form $\sum_{l=0}^{4} c_l v_{i,l}$, where $c_0 \neq 0$. Therefore, (53) and (54) can be solved for the ghost point values $u_{i,0}$ and $v_{i,0}$.

We define the following boundary operators for two-dimensional grid functions,

$$B_{sg1}(\mathbf{u}) := (\mathbf{u}_{1-\tilde{p},j}, \ldots, \mathbf{u}_{0,j}, \mathbf{u}_{N_x+1,j}, \ldots, \mathbf{u}_{N_x+\tilde{p},j})^T, \quad 1-\tilde{p} \le j \le N_y + \tilde{p}, \tag{55}$$

$$B_{sg2}(\mathbf{u}) := \left(\mathbf{u}_{i,N_y+1}, \ldots, \mathbf{u}_{i,N_y+\tilde{p}}\right)^T, \qquad 1-\tilde{p} \le i \le N_x + \tilde{p}. \tag{56}$$

As in the one-dimensional case, $\tilde{p} = \max(2, p)$. The boundary conditions (47) and (48) are discretized by

$$B_{sg1}(\mathbf{u}) = (0, \ldots, 0)^T, \quad B_{sg2}(\mathbf{u}) = (0, \ldots, 0)^T. \tag{57}$$

Using the above notation and the same time discretization as in Section 2.1, we can write the finite difference approximation of the elastic wave equation with super-grid layers on vector form,

$$\rho \frac{\mathbf{u}^{n+1} - 2\mathbf{u}^n + \mathbf{u}^{n-1}}{\Delta_t^2} = \mathbf{L}_h \mathbf{u}^n + \mathbf{f}^n + \frac{\Delta_t^2}{12}\left(\mathbf{L}_h \ddot{\mathbf{u}}^n + \mathbf{f}_{tt}^n\right) - \varepsilon(-1)^p \mathbf{Q}_{2p}\left(\frac{\mathbf{u}_{i,j}^n - \mathbf{u}_{i,j}^{n-1}}{\Delta_t}\right), \tag{58}$$

where $\mathbf{u}^n = (u^n, v^n)^T$ is subject to the normal stress boundary conditions (53), (54) as well as the Dirichlet conditions $B_{sg1}(\mathbf{u}^n) = \mathbf{0}$ and $B_{sg2}(\mathbf{u}^n) = \mathbf{0}$. In (58), the acceleration is defined by

$$\ddot{\mathbf{u}}_{i,j}^n = \left(\mathbf{L}_h \mathbf{u}_{i,j}^n + \mathbf{f}_{i,j}^n\right)/\rho_{i,j}, \quad 1 \le i \le N_x, \ 1 \le j \le N_y, \tag{59}$$

which is subject to the Dirichlet conditions $B_{sg1}(\ddot{\mathbf{u}}^n) = \mathbf{0}$ and $B_{sg2}(\ddot{\mathbf{u}}^n) = \mathbf{0}$. It is also subject to the normal stress conditions (53), (54), where the boundary forcing functions $(\tau^{(xy)}, \tau^{(yy)})$ are replaced by their second time derivatives.

3.2 Energy estimate

In our previous work for second and fourth order accurate methods, e.g., [13, 17], the discrete energy estimate is derived based on the fundamental property

$$(\mathbf{w}, \mathbf{L}_h \mathbf{u})_{hw} = -S_h(\mathbf{w}, \mathbf{u}) + T_h(\mathbf{w}, \mathbf{u}). \tag{60}$$

Here, $(\mathbf{u}, \mathbf{v})_{hw}$ is a weighted scalar product and the bilinear form $S_h(\mathbf{w}, \mathbf{u})$ is symmetric and positive semi-definite. The term $T_h(\mathbf{w}, \mathbf{u})$ is also bilinear and consists of contributions from the boundary. In particular, $T_h(\mathbf{w}, \mathbf{u}) = 0$ when $\mathbf{w}$ satisfies homogeneous Dirichlet conditions, or $\mathbf{u}$ satisfies free surface conditions, see [13, 17] for details.

Our previous estimates hold when the difference operators in $\mathbf{L}_h \mathbf{u}$ are SBP modified at all boundaries of the domain, and when the scalar product is correspondingly weighted near all boundaries. We proceed by proving that the fundamental relation (60) also holds without SBP modifications near the super-grid boundaries. Define the weighted scalar product for real-valued scalar grid functions $u_{i,j}$ and $v_{i,j}$ by

$$(u, v)_{hw} = h^2 \sum_{j=1}^{N_y} \sum_{i=1}^{N_x} \omega_j u_{i,j} v_{i,j}.$$

The corresponding scalar product for real valued vector grid functions $\mathbf{u}_{i,j}$ and $\mathbf{v}_{i,j}$, is

$$(\mathbf{u}, \mathbf{v})_{hw} = \left(u^{(x)}, v^{(x)}\right)_{hw} + \left(u^{(y)}, v^{(y)}\right)_{hw}, \quad \mathbf{u} = \begin{pmatrix} u^{(x)} \\ u^{(y)} \end{pmatrix}, \quad \mathbf{v} = \begin{pmatrix} v^{(x)} \\ v^{(y)} \end{pmatrix}.$$

Because the difference operators are SBP modified only at the boundary $\eta = 0$, the weight in the scalar product, ω_j, only depends on j. Furthermore, it is only different from unity for $1 \leq j \leq 4$.

To handle the relation between cross-terms and second derivatives, we need to show that $D^{(x)}u$ is anti-symmetric.

Lemma 5. *Let $u_{i,j}$ and $v_{i,j}$ be real-valued grid functions satisfying the boundary conditions $B_{sg1}(u) = \mathbf{0}$ and $B_{sg1}(v) = \mathbf{0}$. Let $D^{(x)}$ denote the finite difference operator defined by* (51)*. Then,*

$$\left(v, D^{(x)}u\right)_{hw} = -\left(D^{(x)}v, u\right)_{hw}.$$

Proof. See Appendix A.4. □

To prove an energy estimate for the two-dimensional spatial discretization (58) together with boundary conditions (53), (54), and (57), we proceed as follows. We first apply Lemmas 1, 2, and 5, on each operator in the x-direction. For the operators in the y-direction, Lemmas 1, 2, and 5 are modified by the summation by parts boundary terms at $\eta = 0$, and become

$$\left(v, G^{(y)}(\mu)u\right)_{hw} = -K_0^{(y)}(v,u) - h\sum_{i=1}^{N_x} \mu_{i,1} v_{i,1} B^{(y)} u_{i,1}, \tag{61}$$

$$\left(v, D^{(y)}u\right)_{hw} = -\left(D^{(y)}v, u\right)_{hw} - h\sum_{i=1}^{N_x} u_{i,1} v_{i,1}, \tag{62}$$

$$\left(v, Q_h^{(y)}u\right)_{hw} = C_0^{(y)}(v,u). \tag{63}$$

Here, the function $K_0^{(y)}(v,u)$ contains a sum of one-dimensional functions $K_0(v,u)$, which is defined in Lemma 1. Similarly, the function $C_0^{(y)}(v,u)$ contains a sum of one-dimensional functions $C_0(v,u)$, which is defined in Lemma 2. The super-grid dissipation operator $Q_h^{(y)}$ gives no contributions to the boundary terms at $\eta = 0$, because $\sigma^{(y)}$ is zero there.

Corresponding to lemmas 1 and 2 in the one-dimensional case, the essential properties of the two-dimensional spatial discretization are specified in the following theorem.

Theorem 2. *Let $\mathbf{u}_{i,j}$ and $\mathbf{w}_{i,j}$ be grid functions that satisfy the boundary conditions (53), (54), and* (57)*. The fourth order spatial operators (49), (50) then satisfy*

$$\left(\mathbf{w}, \frac{1}{\phi^{(x)}\phi^{(y)}}\mathbf{L}_h\mathbf{u}\right)_{hw} = -S_h(\mathbf{w},\mathbf{u}), \tag{64}$$

where S_h is bilinear, symmetric, and positive definite. Furthermore, the dissipation operator (52) satisfies

$$\left(\mathbf{w}, \frac{1}{\phi^{(x)}\phi^{(y)}}\mathbf{Q}_{2p}\mathbf{u}\right)_{hw} = C_h(\mathbf{w},\mathbf{u}),$$

where C_h is bilinear, symmetric, and positive semi-definite.

Proof. See Appendix A.5. □

The discretization of the elastic wave equation with super-grid layers, (58), can be written in matrix form as (24), with symmetric positive (semi-)definite matrices K and C_{2p}. Similar to the one-dimensional case, these matrices are defined through $S_h(\mathbf{w},\mathbf{u}) = \mathbf{w}^T K \mathbf{u}$ and $C_h(\mathbf{w},\mathbf{u}) = \mathbf{w}^T C_{2p}\mathbf{u}$. Furthermore, in the two-dimensional case, the matrix Φ is still diagonal, with elements $\phi^{(x)}\phi^{(y)}$. For example,

$$\mathbf{L}_h\mathbf{u} = \phi^{(x)}\phi^{(y)}\left(\frac{1}{\phi^{(x)}\phi^{(y)}}\mathbf{L}_h\mathbf{u}\right) = -\phi^{(x)}\phi^{(y)}K\mathbf{u} = -\Phi K\mathbf{u}.$$

The remaining terms in (58) can be rewritten similarly, allowing the finite difference scheme for the elastic wave equation to be cast in the same matrix formulation as the scalar wave equation, i.e., (24). Theorem 1 therefore applies also to (58), and we obtain our main result.

Theorem 3. *The finite difference scheme (58) with zero forcing* $\mathbf{f}^n = 0$ *and homogeneous boundary conditions (53), (54), and* (57)*, has a non-increasing discrete energy*

$$e^{n+1/2} \le e^{n-1/2} \le \ldots \le e^{1/2}.$$

The discrete energy, corresponding to (26)*, is a norm of the solution when the time step satisfies the inequalities corresponding to (27) and (28). Therefore, the scheme (58) is stable.*

3.3 Time step restriction in several space dimensions

Similar to the one-dimensional wave equation, a von Neumann analysis can be used to estimate the stability limit of the time step for the two-dimensional elastic wave equation. Here we will only study the influence of the super-grid dissipation in the fully discretized elastic wave equation and therefore take $\mathbf{L} = 0$ in (58). After assuming zero forcing, constant stretching and dissipation coefficients as well as material properties, the dissipative terms in (58) become

$$\frac{\mathbf{u}^{n+1} - 2\mathbf{u}^n + \mathbf{u}^{n-1}}{\Delta_t^2} = -\varepsilon(-1)^p\left[\phi_0^{(x)}\sigma_0^{(x)}(D_+^\xi D_-^\xi)^p + \phi_0^{(y)}\sigma_0^{(y)}(D_+^\eta D_-^\eta)^p\right]\left(\frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\Delta_t}\right).$$

To perform the von Neumann analysis, we assume that the solution is 2π-periodic in ξ and η, and expand the solution in a Fourier series. After some algebra, the necessary condition for stability becomes

$$0 \le \gamma_{2p}\left(\phi_0^{(x)}\sigma_0^{(x)} + \phi_0^{(y)}\sigma_0^{(y)}\right) \le \frac{2}{4^p},$$

where γ_{2p} is the scaled dissipation coefficient defined by (35). On sides away from corners, either $\sigma^{(x)} = 0$ or $\sigma^{(y)} = 0$. However, both $\sigma^{(x)}$ and $\sigma^{(y)}$ are positive in the corner regions, where two super-grid layers meet.

To avoid having to significantly reduce γ_{2p} compared to the one-dimensional case, it is necessary to reduce $\sigma^{(x)}$ and $\sigma^{(y)}$ near the corners. A simple solution is provided by introducing a linear taper function. For example, in the corner region $0 \le \xi \le x_1$, $0 \le \eta \le y_1$, we define

$$\tau(x) = \begin{cases} \alpha, & x < 0, \\ \alpha + (1-\alpha)x/\ell, & 0 \le x \le \ell, \\ 1, & x > \ell. \end{cases}$$

We take $\alpha = 1/3$ and define the two-dimensional damping functions by

$$\sigma^{(x)}(\xi,\eta) = \tau(\eta)\sigma(\xi), \quad \sigma^{(y)}(\xi,\eta) = \tau(\xi)\sigma(\eta),$$

where $\sigma(x)$ is the one-dimensional damping function (6). Using this construction, the strength of the damping is determined by

$$I_2(\xi,\eta) := \phi^{(x)}\sigma^{(x)} + \phi^{(y)}\sigma^{(y)} = \tau(\eta)\psi(\xi) + \tau(\xi)\psi(\eta),$$

where $\psi(x)$ is the auxiliary function (5). This construction satisfies $\max I_2 = 1$. Away from the corner, the strength of the damping is the same as in the one-dimensional case because $\psi(x) = 0$ and $\tau(x) = 1$ for $x \geq \ell$. Therefore, $I_2(\xi,\eta) = \psi(\eta)$ for $\xi \geq \ell$ and $I(\xi,\eta) = \psi(\xi)$ for $\eta \geq \ell$. At the corner, $\tau(0) = 1/3$ and $\psi(0) = 1$, giving $I_2(0,0) = 2/3$. The function $I_2(\xi,\eta)$ has a local maxima along the diagonal $\xi = \eta \approx 0.31\ell$, where $I_2 \approx 0.983$. The tapering approach is straight forward to generalize to the other corners of the computational domain.

In three dimensions, the strength of the damping equals $I_3 := \phi^{(x)}\sigma^{(x)} + \phi^{(y)}\sigma^{(y)} + \phi^{(z)}\sigma^{(z)}$. We generalize the tapering approach by defining $\sigma^{(x)}(\xi,\eta,\zeta) = \tau(\eta)\tau(\zeta)\sigma(\xi)$, etc. This construction also satisfies $\max I_3 = 1$. Note that the two-dimensional strength is recovered along edges of the three-dimensional domain (where two super-grid layers meet), because $I_3(\xi,\eta,\zeta) = I_2(\xi,\eta)$ for $\zeta \geq \ell$, etc. In corners where three supergrid layers meet, the strength of the damping has a local maxima along the space-diagonal $\xi = \eta = \zeta \approx 0.37\ell$ where $I_3 \approx 0.823$.

The tapering approach is of significant practical importance in three-dimensional calculations, where up to three super-grid layers can meet at corners. This is because the tapering keeps the maximum strength of the super-grid damping approximately the same along sides, edges, and corners of the computational domain. Let γ_{2p} be the damping coefficient that makes the time stepping stable in the case with super-grid damping in only one direction. With the tapering approach, this value will also work when three super-grid layers meet at a corner. Without the tapering approach, the time stepping would become unstable unless the damping coefficient is reduced to approximately $\gamma_{2p}/3$. Because the maximum strength of the damping is reduced by a factor of three along the sides of the domain (where only one super-grid damping term is active), the layers would need to be approximately three times thicker to damp out the solution to the same level. Since the super-grid layers are added outside the domain of interest, tripling their thickness would significantly increase the total number of grid points in a three-dimensional case, and make the calculation much more expensive.

4 Numerical experiments

All simulations reported here were performed with the open source code SW4, version 1.0 [15], which solves the three-dimensional elastic wave equation on parallel computers. This code implements the three-dimensional version of the numerical methods described in the previous sections, which satisfy corresponding stability and accuracy results. In all numerical experiments, the order of the super-grid dissipation operator will be either 4 or 6, and the threshold value for the super-grid stretching functions is set to $\varepsilon_L = 10^{-4}$. All calculations use a box-shaped computational domain $(x,y,z) \in [0,x_{max}] \times [0,y_{max}] \times [0,z_{max}]$. A free surface boundary condition is imposed along $z = 0$ and super-grid layers are included on all other sides of the domain.

4.1 Lamb's problem

Lamb [8] derived an analytic solution of the elastic wave equation in a homogeneous half-space, subject to an impulsive vertical point forcing applied on the free surface boundary. Many generalizations have been made to Lamb's original derivation, see for example [11] or [5]. Here we focus on the case with $\lambda = \mu$ (Poisson ratio 1/4) where the evaluation of the analytic solution is somewhat simplified.

We shall solve Lamb's problem numerically and take the domain of interest to be $\ell \le x \le 8+\ell$, $\ell \le y \le 8+\ell$, $0 \le z \le 4+\ell$. The forcing is given by the singular point force

$$\mathbf{f}(\mathbf{x},t) = \begin{pmatrix} 0 \\ 0 \\ g(t)\delta(\mathbf{x}-\mathbf{x}_0) \end{pmatrix},$$

where $\delta(\mathbf{x}-\mathbf{x}_0)$ is the Dirac distribution centered at $\mathbf{x}_0 = (4+\ell, 4+\ell, 0)$. The point force is discretized in space by using the technique described in [14]. The time function satisfies

$$g(t) = \begin{cases} 16384\, t^7(1-t)^7, & 0 < t < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{65}$$

The source time function $g(t)$ is six times continuously differentiable, symmetric around $t = 0.5$, where $g(0.5) = 1$. The smoothness in time of the point forcing translates to smoothness in space of the solution after the point force has stopped acting, i.e., for times $t > 1$. Super-grid layers of width ℓ are added to all sides of the domain of interest, except along $z = 0$, where homogeneous free surface conditions corresponding to (53) and (54) are imposed. We choose the units such that the homogeneous elastic material has the properties $\mu = \lambda = \rho = 1$. The computational domain is taken to be $0 \le x \le 8 + 2\ell =: x_{max}$, $0 \le y \le 8 + 2\ell =: y_{max}$, $0 \le z \le 4 + \ell =: z_{max}$.

Figure 3 shows the numerical solution at three different times when the super-grid layer has thickness $\ell = 2$, the grid size is $h = 0.02$, and the fourth order damping coefficient is $\gamma_4 = 0.02$. Here the magnitude of the displacement, $\sqrt{u^2+v^2+w^2}$, is plotted. The top left and right subfigures show a strong Rayleigh surface wave, a shear wave, and the remnants of a weak compressional wave. In this material, the shear wave moves outwards with phase velocity $c_s = 1$ and the compressional wave has phase velocity $c_p = \sqrt{3}$. For a material with $\mu = \lambda = 1$ it can be shown that the Rayleigh surface wave propagates with phase velocity $c_r \approx 0.92$. Because the wave speed in each direction of the super-grid layers is proportional to the value of the corresponding stretching function, the solution slows down and becomes compressed inside the super-grid layers. Also note that the wave fronts tend towards a square shape as time progresseses. We remark that no artificially reflected waves are visible within the domain of interest, here outlined with a dashed line.

Mooney [11] gives explicit expressions for the analytical solution of Lamb's problem on the surface $z = 0$ in terms of a Green's function, $G(t)$. The z-component of the solution at a point on the surface satisfies

$$w(x,y,0,t) = \frac{K}{r}\int_0^t g'(t-\tau)\, G\left(\frac{\tau}{r}\right)\, d\tau, \tag{66}$$

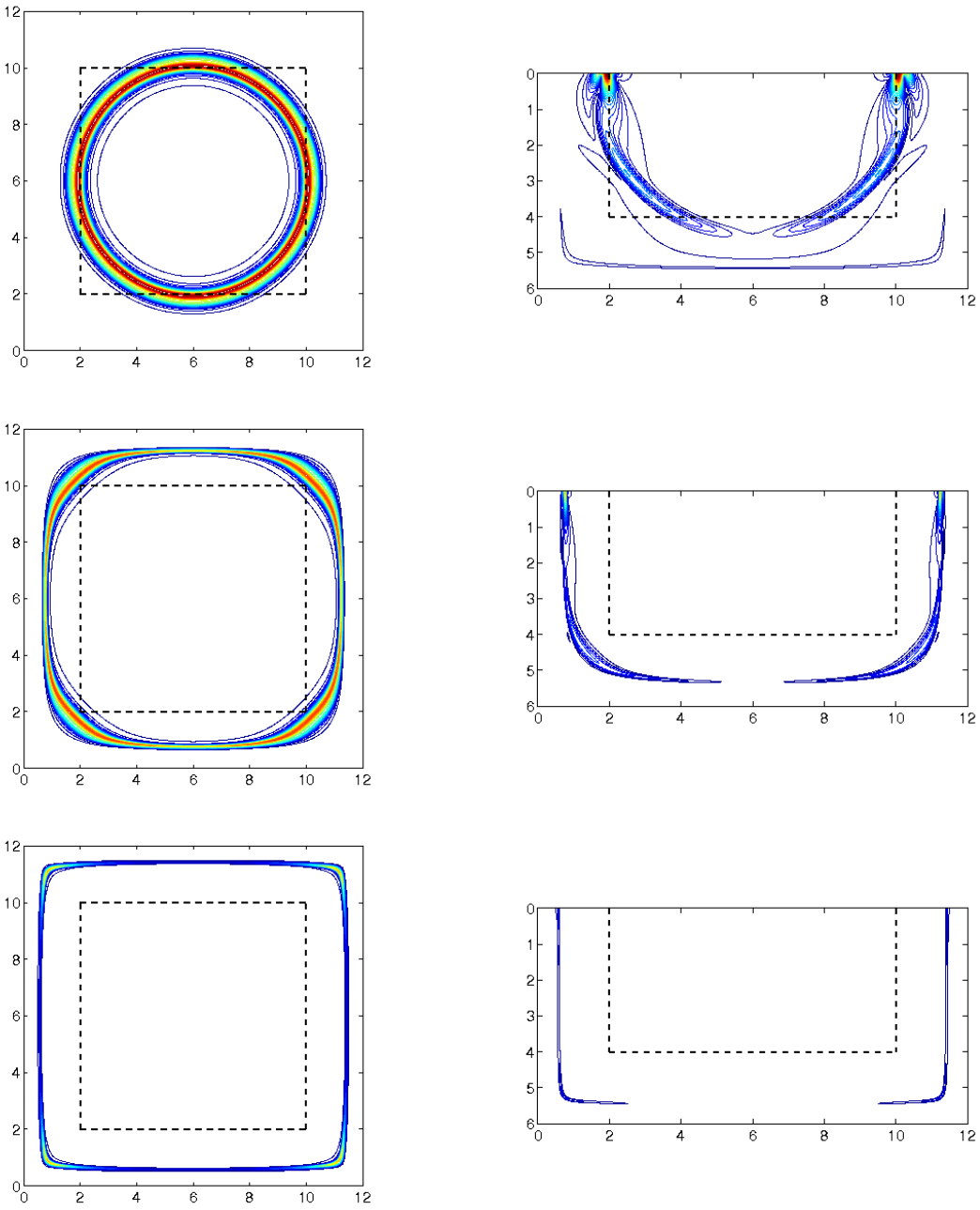


Figure 3: *Magnitude of the displacement for Lamb's problem at times 5, 7, and 9 (top to bottom) in the $z = 0$ plane (left) and the $x = 6$ plane (right). The super-grid layers are outlined by a dashed line and have thickness $\ell = 2$. The contour levels are the same in all plots and are spaced between 0.01 (dark blue) and 0.26 (red) with step size 0.01.*

where $r = \sqrt{x^2 + y^2}$, and

$$G(\xi) = \begin{cases} 0, & \xi < 1/\sqrt{3}, \\ c_1 + c_2/\sqrt{\gamma^2 - \xi^2} + c_3/\sqrt{\xi^2 - b^2} + c_4/\sqrt{\xi^2 - 1/4}, & 1/\sqrt{3} < \xi < 1, \\ c_5 + c_6/\sqrt{\gamma^2 - \xi^2}, & 1 < \xi < \gamma, \\ c_7, & \gamma < \xi. \end{cases} \tag{67}$$

The values of the constants K, c_i, b, and γ are given in [11], with $b \approx 0.563$ and $\gamma \approx 1.0877$. Hence all integrands are non-singular, except in the third case of (67), which has an integrable singularity at $\xi = \gamma$. When g is given by (65), we obtain the exact solution as a sum of terms that either are integrals of polynomials, or have the form

$$\int \frac{P(\xi)}{\sqrt{\xi^2 - a^2}}\, d\xi. \tag{68}$$

where $P(\xi)$ is a polynomial in ξ. Analytical expressions for integrals of the form (68) can be found, but their numerical evaluation is very sensitive to round-off errors, due to the high polynomial order of P. These analytical formulas are therefore inadequate for numerically calculating the exact solution. Instead, we numerically evaluate the convolution integral (66) using the Quadpack library from the Netlib repository [12]. This approach turns out to be much better conditioned, and permits us to evaluate the formula (66) to within approximately 12 decimal places.

Because the analytical solution is only available along the surface ($z = 0$), we study the accuracy of the numerical solution along the surface of the domain of interest, i.e., for $z = 0$, inside the super-grid layers: $\ell \leq x \leq x_{max} - \ell$, $\ell \leq y \leq y_{max} - \ell$.

As can be seen in Figure 3, the solution is dominated by shear and surface waves. The distance between the point force and the closest super-grid layer equals 4, so the shear waves start arriving at the super-grid layer at time $t = 4$. They leave the surface of the domain of interest at time $t = 1 + 4\sqrt{2} \approx 6.65$. The slowest wave in the solution is the surface wave, propagating at phase velocity $c_r \approx 0.92$. The surface waves therefore leave the domain of interest around $t \approx 1 + 4\sqrt{2}/0.92 \approx 7.15$. After that time, the exact solution is zero along the surface of the domain of interest.

Figure 4 shows the L_2 norm of the error in the w-component of the solution, as function of time. The norm is evaluated over the surface of the domain of interest. Note that the point force makes the exact solution unbounded at $\mathbf{x} = \mathbf{x}_0$ for $0 < t < 1$, making the norm of the error undefined. A few grid points near $\mathbf{x}_0$ are therefore excluded from the norm calculation. The errors corresponding to grid sizes $h = 0.04$ (blue), $h = 0.02$ (red), and $h = 0.01$ (black) are shown in Figure 4. Three different regimes of the error can be distinguished. First, for $0 < t < 1$, the point force is active and the numerical solution has a large error near $\mathbf{x}_0$, where the exact solution is unbounded. No reduction of the norm of the error is obtained as the grid is refined. Then follows a time interval where the error first increases and then decreses, i.e., $1 \leq t \leq 7.15$. Here the forcing is zero and the solution error is dominated by propagation errors. Note that the L_2-norm of the error is reduced by approximately a factor of 16 each time the grid size is halved, indicating a fourth order convergence rate. The reason the error decays between $t \approx 5$ and $t \approx 7.15$ is because the shear and surface waves are leaving the domain of interest. Artificial reflections from the super-grid layers become noticeable around $t \approx 7.5$ and we take the third interval to be $t > 7.5$. The simulations are run to time 30.

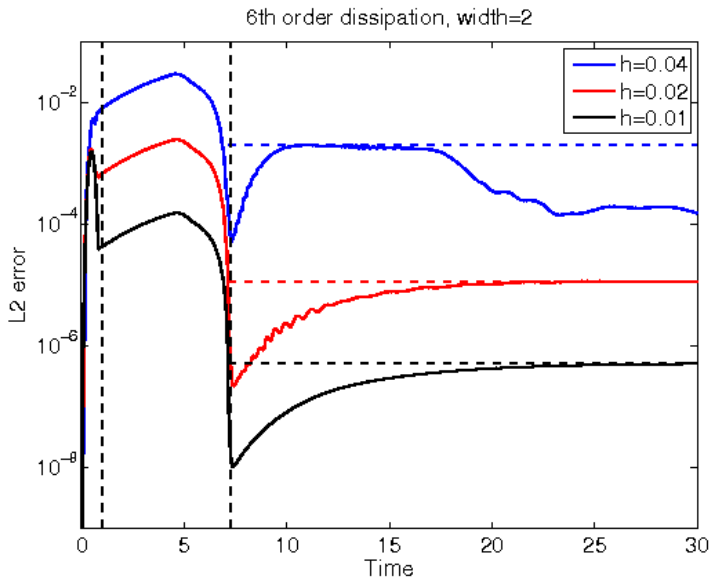


Figure 4: *L_2 error in the vertical component of Lamb's problem with sixth order artificial dissipation for grid sizes $h = 0.04$ (blue), $h = 0.02$ (red), and $h = 0.01$ (black). The width of the super-grid layer is $\ell = 2$. The dashed vertical lines indicate times $t = 1$ and $t = 7.15$. The dashed horizontal lines indicate the max errors in time for $t > 7.5$.*

To investigate the amount of artificial reflections from the super-grid layers, we study the maximum value of the L_2 errors for times $7.5 < t \leq 30$, see Table 1. Here the width of the layer, $\ell = N_{SG}h$, is varied as well as the order of the artificial dissipation. The coefficients for the fourth and six order dissipations are $\gamma_4 = 0.02$ and $\gamma_6 = 0.005$, except for the first entry ($N_{SG} = 13$, $h = 0.04$), where those values lead to numerical instabilities. In this case, stability was regained by reducing the coefficients to $\gamma_4 = 0.01$ and $\gamma_6 = 0.002$, respectively.

The amount of artificial reflections depends strongly on the width of the super-grid layer, ℓ. On the coarsest grid ($h = 0.04$), the fourth order dissipation gives slightly smaller errors than the sixth order dissipation for all widths. However, the sixth order dissipation shows superior performance as the grid is refined. Reflected waves propagate from the layer back into the domain of interest. Because the fourth order dissipation adds a third order perturbation to the elastic wave equation, we can only expect the artificial reflections to decay as $\mathcal{O}(h^3)$ when the fourth order dissipation is used. Based on the same argument, the sixth order dissipation should result in a fifth order perturbation of the elastic wave equation. Because the interior scheme is fourth order accurate, the overall convergence rate should be $\mathcal{O}(h^4)$. Except for the thinnest super-grid layer, the results in Table 1 indicate almost third order convergence for the fourth order dissipation and better than fourth order convergence for the sixth order dissipation.

It is also instructive to compare the sizes of the propagation errors with the reflection errors. In our setup, the propagation error can be quantified as the max value of the L_2 norm of the error during the time interval $1 \leq t \leq 7.5$. Evaluating the errors in the

			4th order diss.		6th order diss.	
Width	N_{SG}	h	max error	ratio	max error	ratio
0.52	13	0.04	1.31e-1	–	1.50e-1	–
0.5	25	0.02	3.33-2	3.93	2.08e-2	7.21
0.5	50	0.01	8.51e-3	3.91	2.04e-3	10.20
1	25	0.04	1.91e-2	–	2.33e-2	–
1	50	0.02	2.73e-3	6.98	8.89e-4	26.14
1	100	0.01	4.88e-4	5.60	4.56e-5	19.48
2	50	0.04	1.13e-3	–	1.97e-3	–
2	100	0.02	1.33e-4	8.49	1.16e-5	170.3
2	200	0.01	1.85e-5	7.17	5.09e-7	22.81

Table 1: *The maximum value of the L_2-norm of the error for times $7.5 < t \leq 30$.*

numerical solution (here denoted by w_h) gives

$$\max_{1\leq t\leq 7.5} \|w(\cdot,\cdot,0,t) - w_h(\cdot,\cdot,0,t)\|_2 \approx \begin{cases} 2.95\cdot 10^{-2}, & h = 0.04, \\ 2.47\cdot 10^{-3}, & h = 0.02, \\ 1.54\cdot 10^{-4}, & h = 0.01. \end{cases}$$

As a minimum requirement, we want the reflection errors from the super-grid layers to be smaller than the propagation errors. While this criteria is satisfied for $\ell = 1$ and $\ell = 2$, it is not satisfied for the thinnest super-grid layer ($\ell = 0.5$). Note that the dominant wave length of the solution is approximately one. Our conjecture is that the super-grid layers need to be at least as wide as this wave length.

It is also interesting to study what happens if a fixed number of grid points are used in the super-grid layer. This means that the width of the layer ℓ becomes smaller as the grid is refined. In Table 1, 50 grid points are used in the layer on line 7 ($h = 0.04$), line 5 ($h = 0.02$) and line 3 ($h = 0.01$). When the fourth order dissipation is used, the error grows monotonically as the grid size is decreased. The results for the sixth order dissipation are only marginally better. Here the error is about the same for $h = 0.04$ and $h = 0.01$, but smaller for $h = 0.02$. We conclude that keeping a fixed number of grid points in the layer leads to a modeling error that does not diminish as the grid size tends to zero.

4.2 A heterogeneous half-space problem with smooth material

To further test the reflection properties of the super-grid approach, we consider a regularized layered material model, where c_p, c_s, and ρ depend on z. We let the compressional

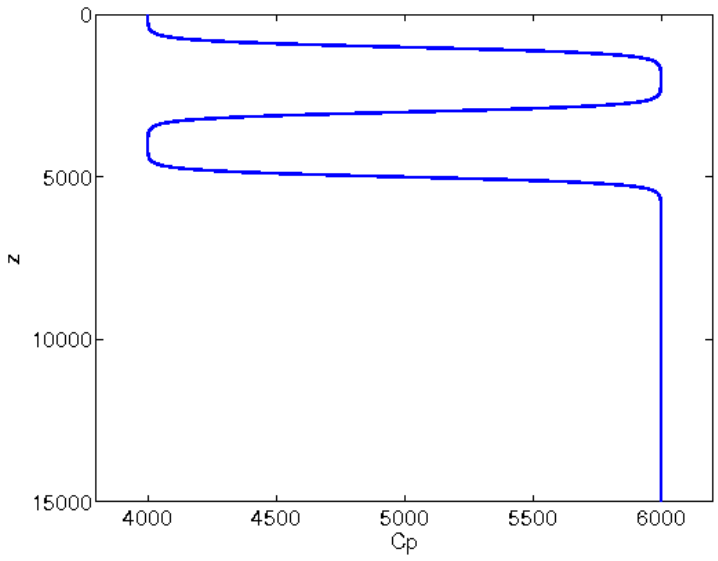


Figure 5: *Compressional velocity (c_p) as function of depth (z) in the vertically layered material.*

wave speed vary between $c_p^{(1)} = 4000$ and $c_p^{(2)} = 6000$,

$$c_p(z) = c_p^{(1)} + \frac{c_p^{(2)} - c_p^{(1)}}{2}\left(1 + \tanh\frac{z - z_1}{L_z}\right) + \frac{c_p^{(1)} - c_p^{(2)}}{2}\left(1 + \tanh\frac{z - z_2}{L_z}\right) + \frac{c_p^{(2)} - c_p^{(1)}}{2}\left(1 + \tanh\frac{z - z_3}{L_z}\right).$$

The transition points are $z_1 = 1000$, $z_2 = 3000$, $z_3 = 5000$, and the transition length scale is $L_z = 200$. The resulting function is plotted in Figure 5. The shear speed and density vary in a corresponding way with $c_s^{(1)} = 2000$, $c_s^{(2)} = 3464$, $\rho^{(1)} = 2600$, and $\rho^{(2)} = 2700$.

The solution is driven by a point moment tensor source,

$$\mathbf{f}(\mathbf{x}, t) = g(t)\begin{pmatrix} 0 & m_{xy} & 0 \\ m_{xy} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}\nabla\delta(\mathbf{x} - \mathbf{x}_s), \tag{69}$$

located at $\mathbf{x}_s = (x_s,\ y_s,\ z_s) = (20,\ 20,\ 2)\cdot 10^3$, with amplitude $m_{xy} = 10^{18}$. The source time function is the Gaussian,

$$g(t) = \frac{1}{2\pi\sigma}e^{(t-t_0)^2/2\sigma^2}, \quad \sigma = 0.12, \quad t_0 = 0.72. \tag{70}$$

We estimate the dominant frequency in the Gaussian by $f_0 = 1/(2\pi\sigma) \approx 1.33$ and the highest significant frequency by $f_{max} \approx 2.5 f_0 \approx 3.32$, which corresponds to a shortest shear wave length of $\min c_s/f_{max} \approx 2000/3.32 \approx 603.2$.

We choose the computational domain to be $(x, y, z) \in [0, 4\cdot 10^4] \times [0, 4\cdot 10^4] \times [0, 1.5\cdot 10^4]$. Super-grid layers of width $5 \cdot 10^3$ are used on all sides, except at $z = 0$, where we impose a free surface boundary condition. As a result, the domain of interest becomes $5 \cdot 10^3 \leq (x, y) \leq 3.5 \cdot 10^4$, and $0 \leq z \leq 10^4$. The simulations are run to time $t = 20$.

In Figure 6 we show snapshots of the magnitude of the numerical solution with grid size $h = 50$ and sixth order dissipation with $\gamma_6 = 5 \cdot 10^{-3}$. The solution is shown along the free surface ($z = 0$) and in the vertical plane $y = 2 \cdot 10^4$. Due to the vertical variation of the material velocity, the solution has much more structure than the solution of Lamb's problem. The source is centered in the fast layer between $z_1 = 1000$ and $z_2 = 3000$ and generates head waves that are transmitted into the slower layers above and below. As the waves propagate further downwards, they speed up again as they enter the fast material for $z > 5000$. Several sets of surface and interface waves can be identified in the solution. We remark that no reflected waves are visible in the domain of interest after time $t \approx 15.5$.

No analytical solution is available for this problem. Instead we assess the convergence rate by comparing solutions on grids of three different grid sizes: $h = 100$, $h = 50$, and $h = 25$. According to the above estimate of the shortest shear wave length, these grid sizes correspond to approximately 6, 12, and 24 grid points per wave length.

We assume that the numerical solution, u_h, is a p^{th} order accurate approximation of the solution of the continuous problem, u, and that the relation

$$u_h \approx u + h^p r, \tag{71}$$

holds, where r is a function that can be bounded independently of the grid size, h. It follows from (71) that $u_{2h} \approx u + 2^p h^p r$ and $u_{4h} \approx u + 4^p h^p r$. Therefore,

$$\frac{\|u_{4h} - u_h\|}{\|u_{2h} - u_h\|} \approx \frac{4^p - 1}{2^p - 1} = 2^p + 1,$$

and we can estimate the convergence rate by $p \approx \log_2(\|u_{4h} - u_h\|/\|u_{2h} - u_h\| - 1)$.

Because it is impractical to store the numerical solution at all points in space and time, we will limit our investigation to study the convergence of the time-dependent solution at fixed locations along the intersection between the free surface, $z = 0$, and the boundary of the domain of interest, $x = 3.5 \cdot 10^4$. For each grid size, we record the solution at seven equally spaced locations between $y_1 = 5 \cdot 10^3$ and the symmetry line $y_7 = 2 \cdot 10^4$,

$$y_k = 5 \cdot 10^3 + (k - 1)2.5 \cdot 10^3, \quad k = 1, 2, \ldots, 7.$$

For example, Figure 7 shows the Cartesian components of the solution as function of time, at the location $(x_4, y_4, z_4) = (3.5, 1.25, 0) \cdot 10^4$. On the right side of Figure 7 we show the difference between the solutions computed with grid sizes $h = 50$ and $h = 25$. Note that the difference is significantly smaller than the solution itself, indicating that it is well-resolved on the grid.

In Table 2 we report the L_2 norm of the differences between the numerical solutions at the seven locations. It is interesting to notice that the numerical solutions seem to converge better near the corner of the domain of interest. For locations y_1-y_4 we observe close to perfect fourth order rate of convergence. The rate for y_5-y_7 is slightly lower and goes down to about 3.3 on the symmetry line $y_7 = 2 \cdot 10^4$.

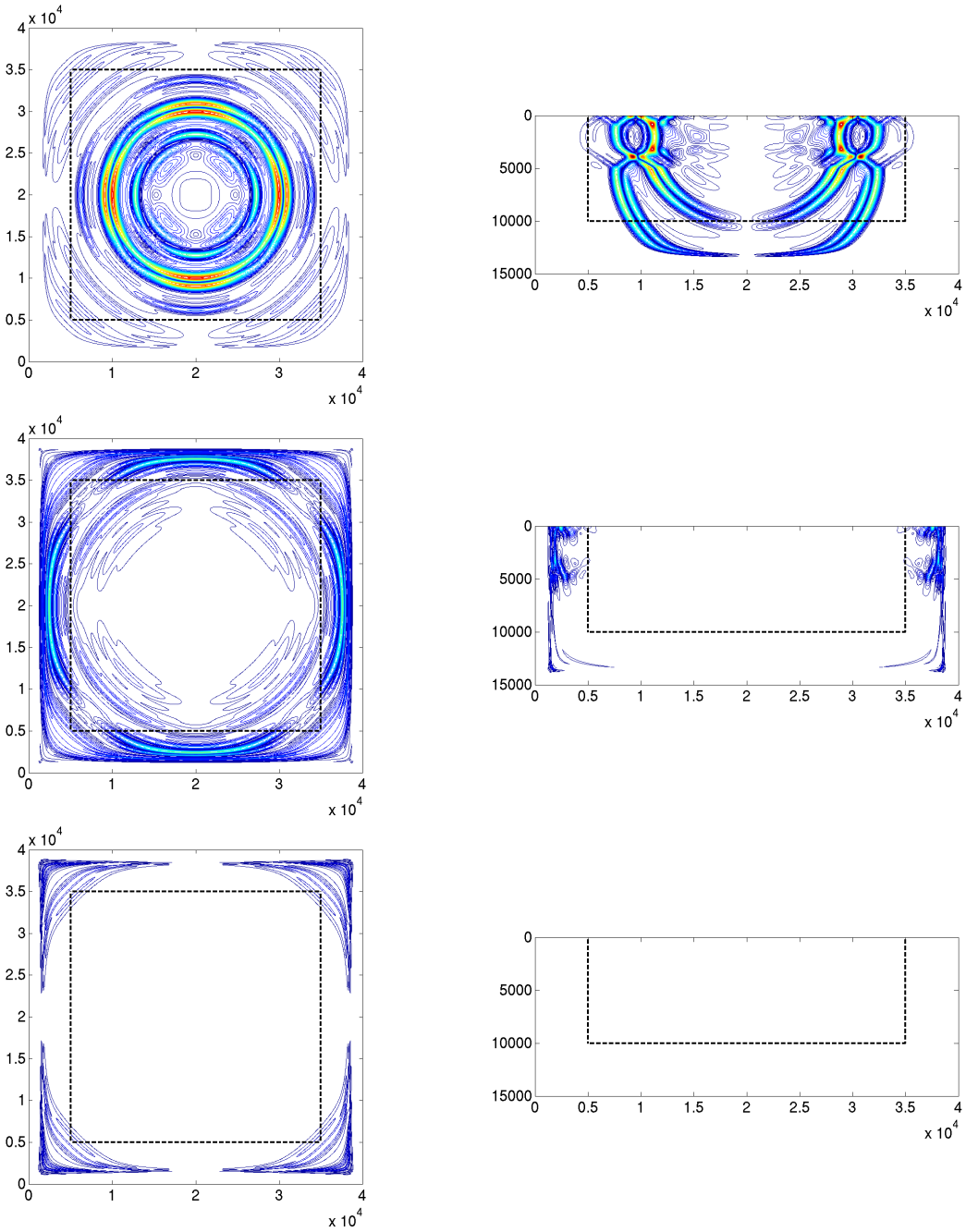


Figure 6: *Magnitude of the displacement in the layered material at times 5.035, 10.07, and 15.105 (top to bottom) in the $z = 0$ plane (left) and the $y = 2 \cdot 10^4$ plane (right). The dashed lines indicate the boundaries of the super-grid layers, which have thickness $\ell = 5 \cdot 10^3$. The contour levels are the same in all plots and are spaced between* 0.05 *(dark blue) and 1.85 (red) with step size 0.05.*

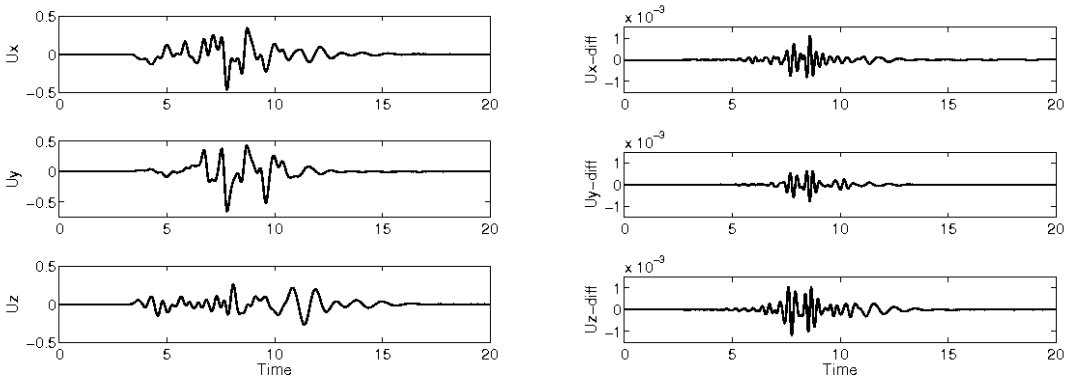


Figure 7: *Solution at* $(x_4, y_4, z_4) = (3.5,\ 1.25,\ 0) \cdot 10^4$ *as function of time, computed with grid size* $h = 50$ *(left). Difference between the numerical solutions computed with grid size* $h = 50$ *and* $h = 25$ *(right).*

Location	y_k	$\|u_{4h} - u_h\|_2$	$\|u_{2h} - u_h\|_2$	ratio	p
1	$0.5 \cdot 10^4$	$4.567 \cdot 10^{-3}$	$2.640 \cdot 10^{-4}$	17.302	4.03
2	$0.75 \cdot 10^4$	$3.767 \cdot 10^{-3}$	$2.188 \cdot 10^{-4}$	17.213	4.02
3	$1 \cdot 10^4$	$3.903 \cdot 10^{-3}$	$2.269 \cdot 10^{-4}$	17.197	4.01
4	$1.25 \cdot 10^4$	$4.284 \cdot 10^{-3}$	$2.538 \cdot 10^{-4}$	16.882	3.99
5	$1.5 \cdot 10^4$	$3.267 \cdot 10^{-3}$	$2.096 \cdot 10^{-4}$	15.580	3.87
6	$1.75 \cdot 10^4$	$2.643 \cdot 10^{-3}$	$2.212 \cdot 10^{-4}$	11.950	3.45
7	$2 \cdot 10^4$	$2.679 \cdot 10^{-3}$	$2.456 \cdot 10^{-4}$	10.906	3.31

Table 2: *The* L_2*-norm of the difference between the numerical solutions at the locations* (x_k, y_k, z_k)*, where* $x_k = 3.5 \cdot 10^4$ *and* $z_k = 0$*. Here,* $p = \log_2(ratio - 1)$*.*

Estimating the convergence rate without access to an analytical solution requires the solution to be sufficiently well resolved on all grids. From the snapshots in Figure 6 we observe that the surface waves are stronger along the symmetry lines than along the diagonal. Since they have slightly shorter wave lengths than the shear waves, it is possible that the surface waves are only marginally resolved on the coarsest grid. This may explain the slightly slower convergence rates near $y = 2 \cdot 10^4$. Nevertheless, we conclude that the super-grid approach is robust and leads to very small artificial reflections, also when the material model is heterogeneous.

5 Conclusions

We have developed a new a finite difference method to approximate the elastic wave equation with super-grid layers. The method combines fourth order accurate summation by parts (SBP) operators [17] with centered fourth order accurate finite difference formulas in the interior of the domain. To make the implementation of the method more efficient and greatly simplified in multi-dimensional domains, our main idea is to only use SBP operators near the physical free surface boundary. The centered finite difference formulas are used all the way up to the artificial super-grid boundaries, where the computational domain is truncated. This approach is made possible by enforcing homogeneous Dirichlet boundary conditions at several grid points outside the super-grid boundaries. Even though the overall discretization does not satisfy the principle of SBP, we have proven by energy estimates that the fully discrete approximation is stable.

One very desirable property of the super-grid method is that, with a wide enough layer, the modeling error from truncating the domain can be made as small as, or smaller than, the wave propagation errors from the interior scheme. This allows the total error in the solution to converge with full order of accuracy as the grid size tends to zero. As shown in the numerical experiments, fourth order accuracy can be achieved with a sixth order artificial damping term, if the super-grid layer has a constant thickness, i.e., independent of the grid size. This thickness can be considerably thinner than the trivial layer, which would be as wide as the distance traveled by the fastest wave over the duration of the entire simulation. However, while the modeling error is reduced by making the super-grid layer thicker, it also increases the computational cost and storage requirements of the simulation. There is therefore a trade-off between computational cost and accuracy of the solution, which is tunable by only changing one parameter, i.e., the width of the super-grid layers.

The super-grid approach can be generalized to curvilinear coordinates. This allows the free surface condition to be imposed on a non-planar surface, which for example is very desirable for modeling seismic wave propagation in the presence of realistic topography. The curvilinear super-grid approach has been implemented as part of the open source code SW4 [15].

The basic finite difference method itself could be generalized from fourth to higher order accuracy. All the summation by parts operators, and the modified equation based time stepping method, are available to at least sixth order of accuracy. The order of the dissipation operator in the super-grid layer would also need to be increased to match a higher order accurate interior scheme. This can be done in a straight forward way by noting that the $2p$ order dissipation operator gives a $2p-1$ order truncation error in the super-

grid layers. Since these errors can propagate into the domain of interest, it is necessary to choose p such that $2p-1$ is larger than or equal to the expected convergence rate of the interior difference scheme. For example, to obtain overall sixth order of accuracy, it would be necessary to use an eight order artificial dissipation operator.

Additional extensions of the current work could include a more detailed analysis of the modeling error from truncating the domain. In particular it would be desirable to establish a mathematical proof of our numerical observation that the solution converges with optimal rate, if the super-grid layer is sufficiently wide.

A Proofs of lemmas and theorems

A.1 Properties of $G(\mu)u$

We simplify the notation by first analyzing the function $K_1(v,u) := (v, G(\mu)u)_{h1}$. The finite difference operator $G(\mu)u_j$, defined by (10), can be written as a sum of three difference operators

$$G(\mu)u_j = D^{(x1)}\left(\mu_j D^{(x1)} u_j\right) + \frac{h^4}{18} D_+ D_- D_+ \left(\tilde{\mu}_j D_- D_+ D_- u_j\right) - \frac{h^6}{144}\left(D_+ D_-\right)^2 \left(\mu_j \left(D_+ D_-\right)^2 u_j\right), \quad (72)$$

where $\tilde{\mu}_j = (\mu_j + \mu_{j-1})/2$. Here, D_+ and D_- denote the standard forward and backward divided difference operators. The term $D^{(x1)}w_j$ is a centered fourth order accurate approximation of $w_\xi(\xi_j)$. It can be written

$$D^{(x1)}w_j := D_0 w_j - \frac{h^2}{6} D_0 D_+ D_- w_j, \quad D_0 = \frac{1}{2}(D_+ + D_-). \quad (73)$$

In the following we set $N = N_x$.

We want to analyze $G(\mu)u_j$ for $j = 1, 2, \ldots, N$. We first comment on the width of the stencil. The terms $D^{(x1)}\mu_j D^{(x1)}u_j$ and $(D+D)^2\mu_j(D+D-)^2 u_j$ are both nine points wide. But the sum of the two is only seven point wide, since the outermost points in the stencils have weights of equal magnitude but opposite signs. Similarly, the sum of these two nine-point stencils and the seven point stencil, $D_+D_-D_+(\tilde{\mu}_j D_- D_+ D_- u_j)$, has zero weights on the outermost terms, making the resulting stencil five points wide.

Since $G(\mu)u_j$ is a five point formula, its values at the interior points $1 \le j \le N$ are only influenced by u_q for $-1 \le q \le N+2$. We next treat the operators term by term. For simplicity, we introduce some additional artificial ghost points. Note that this is not strictly necessary, because $G(\mu)u_j$ is a five point formula, but it simplifies the presentation. The boundary condition $B_{sg}(u) = \mathbf{0}$ sets $u_{-1} = u_0 = 0$ and $u_{N+1} = u_{N+2} = 0$. However, we can impose boundary conditions at additional ghost points without changing $G(\mu)u_j$ for $j = 1, 2, \ldots, N$. In particular, we choose to replace $B_{sg}(u) = \mathbf{0}$ and $B_{sg}(v) = \mathbf{0}$ by imposing homogeneous Dirichlet conditions at four ghost points

$$u_{-3} = u_{-2} = u_{-1} = u_0 = 0, \quad v_{-3} = v_{-2} = v_{-1} = v_0 = 0, \quad (74)$$

$$u_{N+1} = u_{N+2} = u_{N+3} = u_{N+4} = 0, \quad v_{N+1} = v_{N+2} = v_{N+3} = v_{N+4} = 0. \quad (75)$$

It is convenient to analyze $G(\mu)u_j$ by studying each term on the right hand side of (72) independently. We focus on the properties of $G(\mu)u_j$ near the left boundary, and

we extend the grid functions to the semi-infinite domain $j \geq -3$ subject to the boundary conditions (74). We modify the scalar product to be

$$(u, v)_{h0} = h \sum_{j=1}^{\infty} u_j v_j. \tag{76}$$

In this scalar product, the basic forward, backward and centered divided difference operators satisfy the SBP parts identities

$$\begin{aligned} (v, D_+ w)_{h0} &= -(D_- v, w)_{h0} - w_1 v_0, \\ (v, D_- w)_{h0} &= -(D_+ v, w)_{h0} - w_0 v_1, \\ (v, D_0 w)_{h0} &= -(D_0 v, w)_{h0} - \frac{1}{2}(w_0 v_1 + w_1 v_0). \end{aligned} \tag{77}$$

Repeated use of these identities and boundary condition (74) lead to the relations

$$\left(v, D^{(x1)}(\mu D^{(x1)} u)\right)_{h0} = -\left(D^{(x1)} v, \mu D^{(x1)} u\right)_{h0} - J_1, \tag{78}$$

$$(v, D_+ D_- D_+ (\tilde{\mu} D_- D_+ D_- u))_{h0} = -(D_- D_+ D_- v, \tilde{\mu} D_- D_+ D_- u)_{h0} - J_2, \tag{79}$$

$$\left(v, (D_+ D_-)^2 \left(\mu (D_+ D_-)^2 u\right)\right)_{h0} = \left((D_+ D_-)^2 v, \mu (D_+ D_-)^2 u\right)_{h0} + J_3. \tag{80}$$

The boundary terms satisfy

$$\begin{aligned} J_1 &= \frac{1}{144h}(\mu_0 (u_2 - 8u_1)(v_2 - 8v_1) + \mu_{-1} u_1 v_1), \\ J_2 &= \frac{1}{h^5}(\mu_{-1/2} u_1 v_1), \\ J_3 &= \frac{1}{h^7}(\mu_{-1} u_1 v_1 + \mu_0 (u_2 - 4u_1)(v_2 - 4v_1)). \end{aligned}$$

By collecting terms,

$$\begin{aligned} (v, G(\mu)u)_{h0} = -\left(D^{(x1)} v, \mu D^{(x1)} u\right)_{h0} - \frac{h^4}{18}(D_- D_+ D_- v, \tilde{\mu} D_- D_+ D_- u)_{h0} - \\ \frac{h^6}{144}\left((D_+ D_-)^2 v, \mu (D_+ D_-)^2 u\right)_{h0} - J, \end{aligned} \tag{81}$$

where the boundary term satisfies $J = J_1 + h^4 J_2/18 + h^6 J_3/144$. All terms in (81) are symmetric in u and v. Since $\mu > 0$, all terms are negative or zero if $u = v$. The contributions from the right boundary can be analyzed in the same way. Collecting all contributions to $(v, G(\mu)u)_{h1}$ shows that the function K_1 is symmetric, i.e.,

$$K_1(v, u) := -(v, G(\mu)u)_{h1}, \quad K_1(v, u) = K_1(u, v).$$

From the above construction, it is clear that $K_1(u, u) \geq 0$. It remains to show that $K_1(u, u)$ is positive definite, i.e., $K_1(u, u) = 0$ if and only if $u = 0$. Obviously, $K_1(u, u) = 0$ if $u = 0$. Because $K_1(u, u)$ is a sum of non-negative terms, it can only be zero if each term is zero. We choose to study the term

$$T_1(u) := (D_- D_+ D_- u, \tilde{\mu} D_- D_+ D_- u)_{h1} = h \sum_{j=1}^{N} \mu_{j-1/2} (D_- D_+ D_- u_j)^2.$$

The difference equation $D_-D_+D_-u_j = 0$ has the general solution $u_j = \alpha + j\beta + j^2\gamma$ where α, β, and γ are constants. Because $T_1(u)$ only depends on the ghost point values u_{-1}, u_0, and u_{N+1}, the boundary condition $B_{sg}(u) = \mathbf{0}$ gives the linear system

$$\alpha - \beta + \gamma = 0, \tag{82}$$

$$\alpha = 0, \tag{83}$$

$$\alpha + (N+1)\beta + (N+1)^2\gamma = 0. \tag{84}$$

It is straight forward to see that this system only has the trivial solution $\alpha = \beta = \gamma = 0$. We conclude that $T_1(u) = 0$ if and only if $u = 0$. Hence, $K_1(u, u) = 0$ if and only if $u = 0$.

Since $\phi_j = \phi(\xi_j) \geq \varepsilon_L > 0$, the same arguments apply to the function $K_0(v, u) = (v, G(\phi\mu)u)_{h1}$. This proves the lemma.

A.2 The artificial dissipation operator Q_{2p}

We apply the same technique as in section A.1 and start by studying the boundary terms due to the left boundary, using the scalar product (76). For a fourth order dissipation, $p = 2$, and we define $w_j = \sigma_j\rho_j D_+D_-u_j$. We have,

$$(v, Q_4u)_{h0} = (v, D_+D_-w)_{h0}.$$

Combining the first two summation by parts rules in (77) gives

$$(v, D_+D_-w)_{h0} = (D_+D_-v, w)_{h0} - v_0D_-w_1 + w_0D_-v_1.$$

Because v satisfies the boundary condition $B_{sg}(v) = \mathbf{0}$, we have $v_0 = 0$. Therefore, the first boundary term is zero. For the second boundary term we have $D_-v_1 = v_1/h$. It can be further simplified because $B_{sg}(u) = \mathbf{0}$, so $u_{-1} = u_0 = 0$. Therefore, $w_0 = \sigma_0\rho_0u_1/h^2$ and we obtain

$$(v, Q_4u))_{h0} = (v, D_+D_-w)_{h0} = (D_+D_-v, \sigma\rho D_+D_-u)_{h0} + v_1u_1\frac{\sigma_0\rho_0}{h^3}.$$

All terms on the right hand side are symmetric in u and v. Furthermore, they are non-negative when $u = v$. Hence, there is a function $C_0(u, v)$ that does not depend on the ghost point values of u or v, such that

$$(v, Q_4u))_{h0} = C_0(v, u), \quad C_0(u, v) = C_0(v, u), \quad C_0(u, u) \geq 0.$$

The influence of the right boundary can be analyzed in the same way. The same approach applies to all dissipation operators of order $2p$, $p \geq 1$. This proves the lemma.

A.3 One-dimensional energy estimate

Assuming $F(t) = 0$, we derive an energy estimate for (24) by forming the scalar product between $(\bar{u}^{n+1} - \bar{u}^{n-1})\Phi^{-1}$ and (24) (note that Φ is non-singular because $\phi_j \geq \varepsilon_L > 0$). For the left hand side, we get

$$\frac{1}{\Delta_t^2}\left(\bar{u}^{n+1} - \bar{u}^{n-1}, \Phi^{-1}M(\bar{u}^{n+1} - 2\bar{u}^n + \bar{u}^{n-1})\right)_{h1} =$$
$$\frac{1}{\Delta_t^2}\left(\bar{u}^{n+1} - \bar{u}^n, \Phi^{-1}M(\bar{u}^{n+1} - \bar{u}^n)\right)_{h1} - \frac{1}{\Delta_t^2}\left(\bar{u}^n - \bar{u}^{n-1}, \Phi^{-1}M(\bar{u}^n - \bar{u}^{n-1})\right)_{h1}. \tag{85}$$

Because the matrices K, Φ, and M are symmetric, the first two terms on the right hand side of (24) become

$$\left(\bar{u}^{n+1}-\bar{u}^{n-1}, -K\bar{u}^{n}+\frac{\Delta_t^2}{12}KM^{-1}\Phi K\bar{u}^{n}\right)_{h1} = \\ \left(\bar{u}^{n+1}, -K\bar{u}^{n}+\frac{\Delta_t^2}{12}KM^{-1}\Phi K\bar{u}^{n}\right)_{h1} - \left(\bar{u}^{n}, -K\bar{u}^{n-1}+\frac{\Delta_t^2}{12}KM^{-1}\Phi K\bar{u}^{n-1}\right)_{h1}, \quad (86)$$

where we have used $\Phi M^{-1} = M^{-1}\Phi$.

To analyze the dissipative term (last term on the right hand side of (24)), it is helpful to first consider an expression of the type $(\bar{x}+\bar{y}, C\bar{y})_{h1}$, where $C = C_{2p}$. We have

$$(\bar{x}+\bar{y}, C\bar{y})_{h1} = (\bar{x}+\bar{y}, C(\bar{x}+\bar{y}))_{h1} - (\bar{x}+\bar{y}, C\bar{x})_{h1}.$$

Also, $(\bar{x}+\bar{y}, C\bar{y})_{h1} = (\bar{x}, C\bar{y})_{h1} + (\bar{y}, C\bar{y})_{h1}$. Because C is symmetric,

$$(\bar{x}+\bar{y}, C\bar{y})_{h1} = \frac{1}{2}(\bar{x}+\bar{y}, C(\bar{x}+\bar{y}))_{h1} - \frac{1}{2}(\bar{x}+\bar{y}, C\bar{x})_{h1} + \frac{1}{2}(\bar{x}, C\bar{y})_{h1} + \frac{1}{2}(\bar{y}, C\bar{y})_{h1} = \\ \frac{1}{2}(\bar{x}+\bar{y}, C(\bar{x}+\bar{y}))_{h1} - \frac{1}{2}(\bar{x}, C\bar{x})_{h1} + \frac{1}{2}(\bar{y}, C\bar{y})_{h1}.$$

Now take $\bar{x} = \bar{u}^{n+1} - \bar{u}^{n}$ and $\bar{y} = \bar{u}^{n} - \bar{u}^{n-1}$. The expression for the dissipative term in (23) becomes

$$\left(\bar{u}^{n+1}-\bar{u}^{n-1}, C_{2p}\left(\bar{u}^{n}-\bar{u}^{n-1}\right)\right)_{h1} = \frac{1}{2}\left(\bar{u}^{n+1}-\bar{u}^{n-1}, C_{2p}\left(\bar{u}^{n+1}-\bar{u}^{n-1}\right)\right)_{h1} \\ - \frac{1}{2}\left(\bar{u}^{n+1}-\bar{u}^{n}, C_{2p}\left(\bar{u}^{n+1}-\bar{u}^{n}\right)\right)_{h1} + \frac{1}{2}\left(\bar{u}^{n}-\bar{u}^{n-1}, C_{2p}\left(\bar{u}^{n}-\bar{u}^{n-1}\right)\right)_{h1}. \quad (87)$$

By inspection of the three terms (85), (86), and (87), it is natural to define the discrete energy according to (26). After re-arranging the terms of (85), (86), and (87), we arrive at the energy estimate (30).

To analyze the properties of $e^{n+1/2}$, we re-write the terms of (26) that involve K. Because K is symmetric,

$$(\bar{u}^{n+1}, K\bar{u}^{n})_{h1} = \frac{1}{4}(\bar{u}^{n+1}+\bar{u}^{n}, K(\bar{u}^{n+1}+\bar{u}^{n}))_{h1} - \frac{1}{4}(\bar{u}^{n+1}-\bar{u}^{n}, K(\bar{u}^{n+1}-\bar{u}^{n}))_{h1}.$$

The same procedure applies to the terms involving the matrix $KM^{-1}\Phi K$, which also is symmetric because $\Phi M^{-1} = M^{-1}\Phi$. The discrete energy $e^{n+1/2}$ can therefore be grouped into two terms

$$e^{n+1/2} = \left(\bar{u}^{n+1}-\bar{u}^{n}, \left(\frac{1}{\Delta_t^2}\Phi^{-1}M - \frac{1}{4}K + \frac{\Delta_t^2}{48}KM^{-1}\Phi K - \frac{\varepsilon}{2\Delta_t}C_{2p}\right)(\bar{u}^{n+1}-\bar{u}^{n})\right)_{h1} \\ + \left(\bar{u}^{n+1}+\bar{u}^{n}, \left(\frac{1}{4}K - \frac{\Delta_t^2}{48}KM^{-1}\Phi K\right)(\bar{u}^{n+1}+\bar{u}^{n})\right)_{h1}.$$

We have $e^{n+1/2} > 0$ if both terms are positive. By taking $\bar{w} = \bar{u}^{n+1} - \bar{u}^{n}$, we see that the first term is positive if (27) is satisfied. Setting $\bar{w} = \bar{u}^{n+1} + \bar{u}^{n}$ shows that the second term is positive if (28) is satisfied. This completes the proof.

A.4 Anti-symmetry of $D^{(x)}$

We first prove the corresponding lemma for the 1-D operator (73), where u and v are 1-D grid functions satisfying the boundary conditions $B_{sg}(u) = \mathbf{0}$ and $B_{sg}(v) = \mathbf{0}$. By expanding the terms in the scalar product and rearranging them,

$$\begin{aligned}\left(v, D^{(x1)}u\right)_{h1} &= \frac{1}{12}\sum_{i=1}^{N} v_i(u_{i-2} - 8u_{i-1} + 8u_{i+1} - u_{i+2}) \\ &= \frac{1}{12}\left[u_{-1}v_1 + u_1 v_{-1} + u_0(-8v_1 + v_2) + v_0(-8u_1 + u_2)\right] + \\ &\qquad \frac{1}{12}\sum_{i=1}^{N} u_i(-v_{i-2} + 8v_{i-1} - 8v_{i+1} + v_{i+2}) + \\ &\frac{1}{12}\left[-u_{N+2}v_N - v_{N+2}u_N + u_{N+1}(8v_N - v_{N-1}) + v_{N+1}(8u_N - u_{N-1})\right]. \quad (88)\end{aligned}$$

The boundary terms are equal to zero because $B_{sg}(u) = \mathbf{0}$ and $B_{sg}(v) = \mathbf{0}$ imply $u_{-1} = u_0 = v_{-1} = v_0 = 0$ and $u_{N+2} = u_{N+1} = v_{N+2} = v_{N+1} = 0$. Hence, we obtain

$$\left(v, D^{(x1)}u\right)_{h1} = -\left(D^{(x1)}v, u\right)_{h1}.$$

Trivial generalizations extend the proof to two-dimensional grid functions.

A.5 Symmetry of the two-dimensional discretization

First, we need the following refinement of (61),

$$\left(v, G^{(y)}(\mu)u\right)_{hw} = -\left(D^{(y)}v, \mu D^{(y)}u\right)_{hw} - \left(v, P^{(y)}(\mu)u\right)_{h} - h\sum_{i=1}^{N_x}\mu_{i,1}v_{i,1}B^{(y)}u_{i,1}, \quad (89)$$

which was proven in [17]. Here $P^{(y)}(\mu)$ is an operator acting in the y-direction, which is positive definite in the un-weighted scalar product $(u, v)_h$,

$$(u, v)_h = h^2\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u_{i,j}v_{i,j}.$$

For details, see [17]. The identity corresponding to (89) for the operator in the x-direction does not have a boundary term. The proof is a trivial generalization of the result in Appendix A.1.

To prove (64), we introduce the grid functions $\mathbf{u} = (u^{(x)},\ u^{(y)})^T$, $\mathbf{w} = (w^{(x)},\ w^{(y)})^T$, and write out the components of (64) as

$$\left(\mathbf{w}, \frac{1}{\phi^{(x)}\phi^{(y)}}\mathbf{L}_h\mathbf{u}\right)_{hw} = \left(w^{(x)}, \frac{1}{\phi^{(x)}\phi^{(y)}}L_h^{(u)}\mathbf{u}\right)_{hw} + \left(w^{(y)}, \frac{1}{\phi^{(x)}\phi^{(y)}}L_h^{(v)}\mathbf{u}\right)_{hw}. \quad (90)$$

The first term on the right hand side is expanded as

$$\left(w^{(x)}, \frac{1}{\phi^{(x)}\phi^{(y)}} L_h^{(u)} \mathbf{u}\right)_{hw} = \left(w^{(x)}, \frac{1}{\phi^{(y)}} G^{(x)} \left(\phi^{(x)}(2\mu+\lambda)\right) u^{(x)}\right)_{hw} + \left(w^{(x)}, D^{(x)}\lambda D^{(y)} u^{(y)}\right)_{hw} + \left(w^{(x)}, D^{(y)}\mu D^{(x)} u^{(y)}\right)_{hw} + \left(w^{(x)}, \frac{1}{\phi^{(x)}} G^{(y)} \left(\phi^{(y)}\mu\right) u^{(x)}\right)_{hw}, \quad (91)$$

where we have used that $\phi^{(x)}$ does not depend on η_j, and $\phi^{(y)}$ does not depend on ξ_i. Next the summation by parts identites are used on each term in (91). As shown in Lemmas 1, 2, and 5, there are no boundary terms from operators in the x-direction. The y-direction formulas are given in equations (62), (63), and (89). The resulting expression is

$$\begin{aligned}\left(w^{(x)}, \frac{1}{\phi^{(x)}\phi^{(y)}} L_h^{(u)} \mathbf{u}\right)_{hw} = &- \left(D^{(x)} w^{(x)}, \frac{\phi^{(x)}}{\phi^{(y)}}(2\mu+\lambda) D^{(x)} u^{(x)}\right)_{hw} - \\ &\left(w^{(x)}, \frac{1}{\phi^{(y)}} P^{(x)}(\phi^{(x)}(2\mu+\lambda)) u^{(x)}\right)_h - \left(D^{(x)} w^{(x)}, \lambda D^{(y)} u^{(y)}\right)_{hw} - \\ &\left(D^{(y)} w^{(x)}, \mu D^{(x)} u^{(y)}\right)_{hw} - \left(D^{(y)} w^{(x)}, \frac{\phi^{(y)}}{\phi^{(x)}} \mu D^{(y)} u^{(x)}\right)_{hw} - \\ &\left(w^{(x)}, \frac{1}{\phi^{(x)}} P^{(y)}(\phi^{(y)}\mu) u^{(x)}\right)_h - h \sum_{i=1}^{N_x} w_{i,1}^{(x)} \frac{\mu_{i,1}}{\phi_i^{(x)}} \left(\phi_i^{(x)} D^{(x)} u_{i,1}^{(y)} + \phi_1^{(y)} B^{(y)} u_{i,1}^{(x)}\right). \quad (92)\end{aligned}$$

By performing a similar expansion of the second term on the right hand side of (90), adding together the results, and completing the squares, we arrive at the final expression

$$\left(\mathbf{w}, \frac{1}{\phi^{(x)}\phi^{(y)}} \mathbf{L}_h \mathbf{u}\right)_{hw} = E_h + P_h + T_h,$$

where

$$\begin{aligned}E_h = &- \left(\phi^{(x)} D^{(x)} w^{(x)} + \phi^{(y)} D^{(y)} w^{(y)}, \frac{\lambda}{\phi^{(x)}\phi^{(y)}} [\phi^{(x)} D^{(x)} u^{(x)} + \phi^{(y)} D^{(y)} u^{(y)}]\right)_{hw} - \\ &\left(\phi^{(y)} D^{(y)} w^{(x)} + \phi^{(x)} D^{(x)} w^{(y)}, \frac{\mu}{\phi^{(x)}\phi^{(y)}} [\phi^{(y)} D^{(y)} u^{(x)} + \phi^{(x)} D^{(x)} u^{(y)}]\right)_{hw} - \\ &\left(\phi^{(x)} D^{(x)} w^{(x)}, \frac{2\mu}{\phi^{(x)}\phi^{(y)}} [\phi^{(x)} D^{(x)} u^{(x)}]\right)_{hw} - \left(\phi^{(y)} D^{(y)} w^{(y)}, \frac{2\mu}{\phi^{(x)}\phi^{(y)}} [\phi^{(y)} D^{(y)} u^{(y)}]\right)_{hw}, \quad (93)\end{aligned}$$

and

$$\begin{aligned}P_h = &- \left(w^{(x)}, \frac{1}{\phi^{(y)}} P^{(x)}(\phi^{(x)}(2\mu+\lambda)) u^{(x)}\right)_h - \left(w^{(x)}, \frac{1}{\phi^{(x)}} P^{(y)}(\phi^{(y)}\mu) u^{(x)}\right)_h - \\ &\left(w^{(y)}, \frac{1}{\phi^{(y)}} P^{(x)}(\phi^{(x)}\mu) u^{(y)}\right)_h - \left(w^{(y)}, \frac{1}{\phi^{(x)}} P^{(y)}(\phi^{(y)}(2\mu+\lambda)) u^{(y)}\right)_h. \quad (94)\end{aligned}$$

The boundary terms are

$$T_h = -h\sum_{i=1}^{N_x} w_{i,1}^{(y)} \frac{1}{\phi_i^{(x)}} \left(\phi_i^{(x)} \lambda_{i,1} D^{(x)} u_{i,1}^{(x)} + \phi_1^{(y)} (2\mu_{i,1} + \lambda_{i,1}) B^{(y)} u_{i,1}^{(y)} \right) - h\sum_{i=1}^{N_x} w_{i,1}^{(x)} \frac{\mu_{i,1}}{\phi_i^{(x)}} \left(\phi_i^{(x)} D^{(x)} u_{i,1}^{(y)} + \phi_1^{(y)} B^{(y)} u_{i,1}^{(x)} \right), \quad (95)$$

which vanish under the homogeneous boundary condition (53)–(54), because $\phi_1^{(y)} = 1$. Hence, we have

$$S_h(\mathbf{w}, \mathbf{u}) = -E_h - P_h.$$

Here E_h is an approximation of the spatial terms in the elastic energy (46), and P_h is symmetric in its arguments and positive definite.

The dissipation operator can similarly be expanded into the four terms

$$\left(\mathbf{w}, \frac{1}{\phi^{(x)}\phi^{(y)}} \mathbf{Q}_{2p}(\mathbf{u}) \right)_{hw} = \left(w^{(x)}, \frac{1}{\phi^{(y)}} Q_{2p}^{(x)} (\sigma^{(x)} \rho) u^{(x)} \right)_{hw} + \left(w^{(x)}, \frac{1}{\phi^{(x)}} Q_{2p}^{(y)} (\sigma^{(y)} \rho) u^{(x)} \right)_{hw} + \left(w^{(y)}, \frac{1}{\phi^{(y)}} Q_{2p}^{(x)} (\sigma^{(x)} \rho) u^{(y)} \right)_{hw} + \left(w^{(y)}, \frac{1}{\phi^{(x)}} Q_{2p}^{(y)} (\sigma^{(y)} \rho) u^{(y)} \right)_{hw}. \quad (96)$$

By applying Lemma 2 to each term, we see that the expression is symmetric and positive semi-definite. Note that although Lemma 2 holds in the unweighted norm, and (96) uses the weighted norm, there is no difficulty because the y-direction operators in (96) are zero near the boundary $\eta = 0$, where the norm is weighted.

References

[1] D. Appelö and T. Colonius. A high order super-grid-scale absorbing layer and its application to linear hyperbolic systems. *J. Comput. Phys.*, 228:4200–4217, 2009.

[2] E. Bécache, S. Fauqueux, and P. Joly. Stability of perfectly matched layers, group velocities and anisotropic waves. *J. Compt. Phys.*, 188:399–433, 2003.

[3] J. P. Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.*, 114:185–200, 1994.

[4] R. Clayton and B. Engquist. Absorbing boundary conditions for acoustic and elastic wave equations. *Bull. Seismo. Soc. Amer.*, 67, 1977.

[5] A. Cemal Eringen and Erdoğan S. Şuhubi. *Elastodynamics, Volume II.* Elsevier, 1975.

[6] R. L. Higdon. Radiation boundary conditions for elastic wave propagation. *SIAM J. Numer. Anal.*, 27:831–870, 1990.

[7] H.-O. Kreiss and J. Lorenz. *Initial-Boundary Value Problems and the Navier-Stokes Equations.* Academic Press, 1989.

[8] Horace Lamb. On the propagation of tremors over the surface of an elastic solid. *Phil. Trans. Roy. Soc. London, Ser. A*, 203, 1904.

[9] K. Mattsson, F. Ham, and G. Iaccarino. Stable and accurate wave-propagation in discontinuous media. *J. Comput. Phys.*, 227:8753–8767, 2008.

[10] K. Mattsson and J. Nordström. Summation by parts operators for finite difference approximations of second derivatives. *J. Comput. Phys.*, 199:503–540, 2004.

[11] Harold M. Mooney. Some numerical solutions for Lamb's problem. *Bull. Seismo. Soc. Amer.*, 64, 1974.

[12] Netlib. Repository of scientific computing software. `http://www.netlib.org`.

[13] N. A. Petersson and B. Sjögreen. An energy absorbing far-field boundary condition for the elastic wave equation. *Comm. Comput. Phys.*, 6:483–508, 2009.

[14] N. A. Petersson and B. Sjögreen. Stable grid refinement and singular source discretization for seismic wave simulations. *Comm. Comput. Phys.*, 8(5):1074–1110, November 2010.

[15] N. A. Petersson and B. Sjögreen. User's guide to SW4, version 1.0. Technical Report LLNL-SM-642292, Lawrence Livermore National Laboratory, 2013. (Source code available from `computation.llnl.gov/casc/serpentine`).

[16] B. Sjögreen and N. A. Petersson. Perfectly matched layers for Maxwell's equations in second order formulation. *J. Comput. Phys.*, 209:19–46, 2005.

[17] B. Sjögreen and N. A. Petersson. A fourth order accurate finite difference scheme for the elastic wave equation in second order formulation. *J. Sci. Comput.*, 52:17–48, 2012. DOI 10.1007/s10915-011-9531-1.

[18] E. A. Skelton, S. D. M. Adams, and R. V. Craster. Guided elastic waves and perfectly matched layers. *Wave Motion*, 44:573–592, 2007.